SERIES 6000

COMPUTER SYSTEM

REFERENCE MANUAL

Datacraft Corporation

RM61305-00

SERIES 6000
COMPUTER SYSTEM
REFERENCE MANUAL

July, 1970

Datacraft Corporation

1200 N.W. 70th Street   P.O. Box 23550   Fort Lauderdale, Florida 33307   Phone (305) 933-2651

LIST OF EFFECTIVE PAGES

TOTAL NUMBER OF PAGES IN THIS PUBLICATION IS 210
CONSISTING OF THE FOLLOWING:

Page No.	Change No.	Page No.	Change No.	Page No.	Change No.
Title	Original				
A	Original				
i - vi	Original				
1-1 - 1-7	Original				
2-1 - 2-22	Original				
3-1 - 3-136	Original				
4-1 - 4-11	Original				
5-1 - 5-26	Original				

Upon receipt of the second and subsequent changes to this technical document, personnel responsible for maintaining this publication in current status will ascertain that all previous changes have been received and incorporated. Action should be taken promptly if the publication is incomplete.

* The asterisk indicates pages changed, added, or deleted by the current change.

CONTENTS

CONTENTS (Cont'd.)

CONTENTS (Cont'd.)

CONTENTS (Cont'd.)

ILLUSTRATIONS

TABLES

SECTION I
INTRODUCTION

1-1 BASIC SYSTEM DESCRIPTION

The DC 6024 is a medium-scale, general-purpose, digital computer featuring a full cycle time of 600 nanoseconds. A modified version, the DC 6024/3, has a 1 microsecond full cycle time. Both models have a fixed word length of 24 bits. These and other features provide rapid memory access, ease of programming, and a real-time control capability approaching that of a multi-processor or large-scale system.

The DC 6024 and 6024/3 use a magnetic core memory, a multi-access bus structure, and fully-buffered input/output channels. Two's complement arithmetic is performed on parallel, fixed-point, binary operands. Floating-point arithmetic hardware is optionally available on the DC 6024.

Hardware multiply/divide/square root, four priority interrupt levels, and a basic software package are furnished as part of the basic system. Five 24-bit general-purpose registers, three of which may be used for indexing, are included. The DC 6024 is supplied with an ASR-33 console typewriter.

Memory capacity in the basic system is 8,192 words. Memory may be expanded to 65,536 words in 8,192 word increments.

Memory may be accessed at the word, double word and byte levels. In memory configurations employing more than 32,768 words, memory is divided into two sectors, or "maps". The addressing technique permits up to 32,768 words per map to be directly addressed and allows up to 65,536 words to be accessed, irrespective of the maps, by multilevel indirect and index references.

The instruction set includes in excess of 500 discrete operations, providing the programmer with considerable convenience and flexibility. The instruction repertoire features an extensive list of operand, register-to-register and byte manipulations, a versatile set of arithmetic and logical operations and total control over external interrupts and input/output functions.

1-2 OPTIONAL FEATURES

1-2.1 Program Restrict System with Instruction Trap

The Program Restrict system allows areas of memory to be defined under program control and protected from unauthorized access. The Instruction Trap provides a means for preventing the execution of certain predefined instructions when the system is in a restricted environment.

1-2.2 Stall Alarm

The Stall Alarm option provides a method for detecting and correcting a stall condition in the Central Processing Unit (CPU).

1-2.3 Interval Timer

The Interval Timer is an internal CPU timer that provides a method for regulating time segments for operating programs and recording other time intervals through the use of the CPU clock. The Interval Timer is under program control and clocks either CPU time or real-time, depending on the instruction used for timer activation. In addition to its timing applications, the Interval Timer provides the user with an additional 24-bit register that may be accessed through the standard instruction set.

1-2.4 Power Failure Shutdown and Restart

This option provides protection for operating programs in the event of a power failure and restoration when power levels return to normal.

1-2.5 Address Trap

The Address Trap is an on-line debugging aid for use in applications such as breakpoint tracing. An address may be defined under program control so that any time that address is referenced, an interrupt will be generated at the assigned executive level. The Address Trap may be disabled under program control.

1-2.6 Bit Processor

The Bit Processor option allows true bit manipulation of data through single word instructions. A bit may be changed, stored, tested, etc. Seventeen instructions are added to the basic instruction set and two additional registers are included with the option.

1-2.7 Automatic Block Controller (ABC)

The ABC provides high-speed, fully-buffered, data transfer between peripheral devices and memory without program intervention. Up to 14 ABCs can be included in either DC 6024 system.

1-2.8 Scientific Arithmetic Unit (SAU)

The SAU provides concurrent floating-point arithmetic capability independent of the CPU.

1-2.9 Priority Interrupts

Up to eight executive traps and up to 72 levels of external interrupts may be provided with the DC 6024. The DC 6024/3 can accommodate eight executive traps up to 24 external levels. The executive traps are associated with the CPU options and are hardware-controlled. The external interrupts are provided in increments of four levels each and are totally under program control.

1-3 PERIPHERAL UNITS

1-3.1 Typewriters

An ASR-33 console typewriter (Datacraft Model 6001-1) is supplied as standard equipment with the Series 6000 Computer. Remote typewriters available are ASR-35 (Model 6001-2) and KSR-35 (Model 6001-3). All ASR models are equipped with an integral paper tape punch (10 characters per second) and paper tape reader (20 characters per second).

1-3.2 High-Speed Paper Tape Reader

The Datacraft Model 6002-20 paper tape reader operates at 600 characters per second and is equipped with an 8-bit I/O channel. An additional paper tape reader, Model 6002-10 operates at 300 characters per second. It too requires an 8-bit I/O channel.

1-3.3 High-Speed Paper Tape Punch

The Datacraft Model 6003 paper tape punch operates at 110 characters per second. An 8-bit channel is required.

1-3.4 Line Printer

The Model 6004-3 line printer prints at a speed of 1000 lines per minute. A 24-bit I/O channel is required and a character disassembly buffer is included.

1-3.5 Card Reader

Two card reader models, 6005-1 and 6005-2, are available These models operate at speeds of 400 and 1000 cards per minute, respectively. Each model requires a 24-bit I/O channel.

1-3.6 Disc Storage System

The Datacraft Models 6006-2 and 6006-3 Disc Storage Units provide random access bulk storage capacities of 7.25M bytes and 29M bytes, respectively. Maximum transfer rates to and from the DC 6024 CPU are 156,000 bytes per second (-2) and 312,000 bytes per second (-3). A 24-bit ABC I/O channel is required.

1-4 SOFTWARE

Series 6000 software includes a modular resident operating system with interrupt-oriented I/O and a link loader, a basic assembler, and utility and FORTRAN support libraries. Optional software includes a compiler, disc operating system, and associated operating systems. For detailed descriptions of available software, refer to the Series 6000 Standard Software Reference Manual.

1-5 CHARACTERISTICS

The major operating characteristics of the Series 6000 computer system are outlined in the following list. A block diagram of the 6000 system configuration is shown in Figure 1-1.

Memory Capacity .... 8,192 words (basic system)

Memory Expansion .... 65,536 words in 8,192 word increments

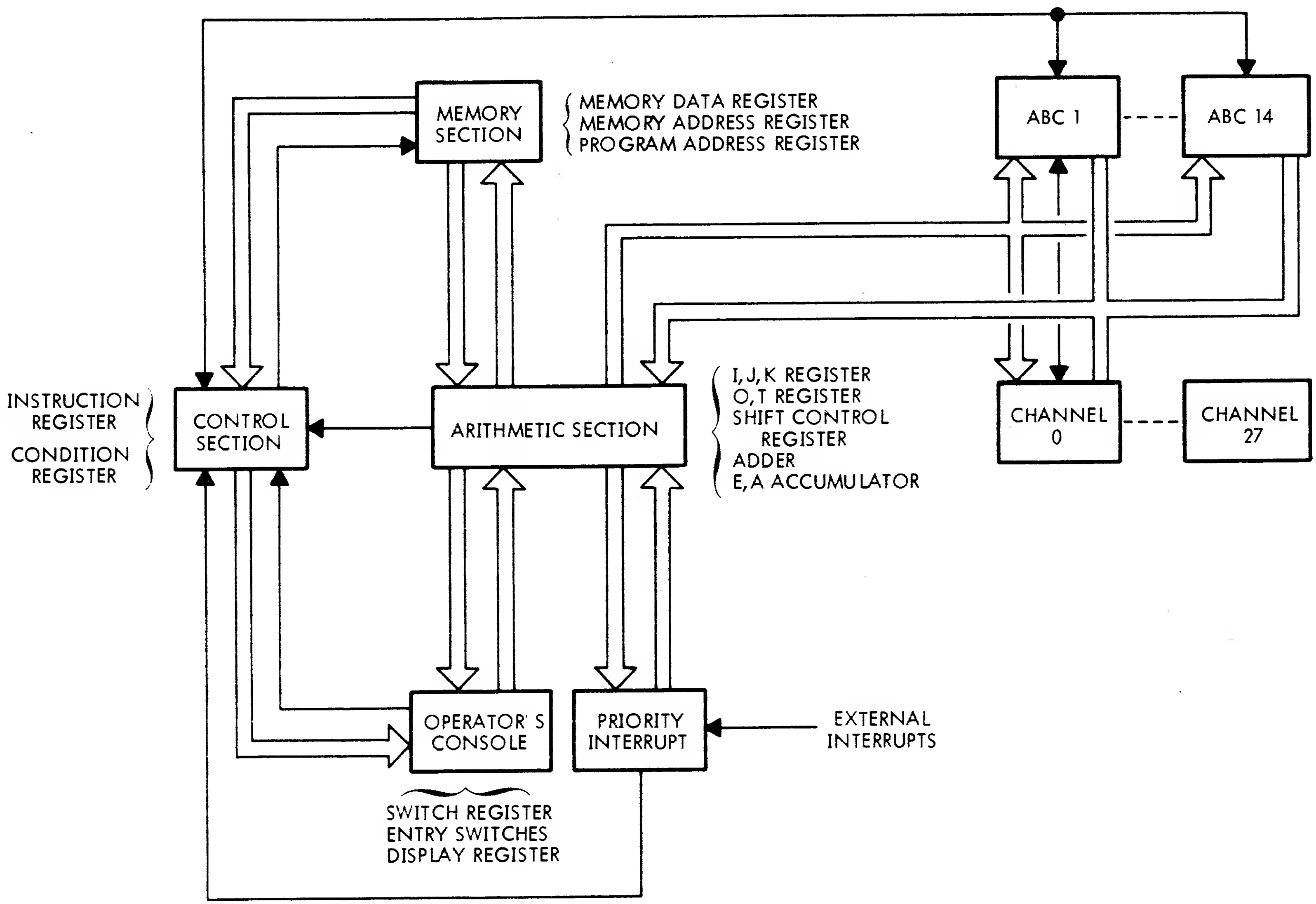
Word Size .... 24 bits plus memory parity

Registers (standard) .... 5 24-bit general registers;
3 may be used for indexing

Addressing Modes:
- Direct addressing .... 32,768 words per map
- Multilevel indirect and index addressing .... 65,536 words

Cycle Time:
- DC 6024 .... 600 nanoseconds
- DC 6024/3 .... 1 microsecond

Execution Times:
- Register-to-register manipulation:
  - DC 6024........................ 600 nanoseconds
  - DC 6024/3 .... 1 microsecond
- Single word I/O:
  - DC 6024 .... 600 nanoseconds
  - DC 6024/3 .... 1 microsecond
- Load/store word:
  - DC 6024 .... 1.2 microseconds
  - DC 6024/3 .... 2 microseconds
- Add, Subtract:
  - DC 6024 .... 1.2 microseconds
  - DC 6024/3 .... 2 microseconds
- Compare:
  - DC 6024 .... 1.2 microseconds
  - DC 6024/3 .... 2 microseconds
- Multiply:
  - DC 6024 .... 4.8 microseconds
  - DC 6024/3 .... 8 microseconds
- Divide:
  - DC 6024 .... 9.0 microseconds
  - DC 6024/3 .... 15 microseconds
- Square Root:
  - DC 6024 .... 8.4 microseconds
  - DC 6024/3 .... 14 microseconds



BD60-001-770

Figure 1-1. Block Diagram - Series 6000 Computer

Block Transfer Capability (Optional ABC):

ABCs per system:	
DC 6024	14 maximum
DC 6024/3	14 maximum
Maximum transfer rate:	
DC 6024	1,666,666 words per second 5,000,000 bytes per second
DC 6024/3	1,000,000 words per second 3,000,000 bytes per second
ABC Setup time:	
DC 6024	1.2 microseconds
DC 6024/3	2 microseconds
Cycles per word transferred	1 (both models)

Priority Interrupt System:

Executive traps	8 (7 reserved)
External levels (program controlled):	
DC 6024	4 (standard) to 72 in 4-level increments
DC 6024/3	4 (standard) to 24 in 4-level increments

Sense Switches	4 (standard)
Control Switches (Switch Register)	24 (standard)

Power Failure Protection:

Memory Data Retention (standard)
Power Failure Shutdown and Restart (optional)

Input Power Requirements	117 VAC ± 10%, 60 Hz

Operating Environment:

Temperature	15° to 45°C (59° to 113°F)
Relative Humidity	10% to 90%

CPU Options:

Program Restrict/Instruction Trap
Stall Alarm
Interval Timer
Power Failure Shutdown and Restart
Address Trap
Scientific Arithmetic Unit (SAU)
Bit Processor
Hardware Bootstrap
Automatic Block Controller (ABC)
External Interrupts

Peripheral Units:

ASR-33 Keyboard/Printer (standard on Series 6000	reader - 20 characters/second; keyboard, punch, prinrer - 10 characters/second
Remote Typewriters:	
ASR-35 Keyboard/Printer	reader - 20 characters/second; keyboard, punch, printer - 10 characters/second
KSR-35 Keyboard/Printer	keyboard, printer - 10 characters/second
Paper Tape Reader	600 characters/second
Paper Tape Punch	110 characters/second
Line Printer	1000 lines/minute
Card Readers:	
Model 6005-1	225 cards/minute
Model 6005-2	1000 cards/minute
Disc Storage Systems:	
Model 6006-2	7.25 M bytes
Model 6006-3	29 M bytes

Software:

Resident Operating System
Basic Assembler
Utility Package
Cross Reference Program
Mainframe Checkout Package
Mainframe Diagnostic Package
FORTRAN Support Library
ASA Standard FORTRAN Compiler (optional)
Object Time Trace Program (optional)
Disc Operating System (optional)

SECTION II
CENTRAL PROCESSING UNIT

2-1 GENERAL DESCRIPTION

The Series 6000 Central Processing Unit (CPU) is a single address, fully parallel, 24-bit word-oriented, stored-program unit. Operations performed by the CPU include data transfers, arithmetic computation, and logical manipulation. These operations are defined by instructions retrieved from memory. The specified operation is performed on single word, double word, or byte operands stored in memory or contained in one of the CPU's five registers. Data word formats, as defined by both hardware and software are illustrated in Figure 2-1.

In addition to the previously mentioned registers, the CPU contains an Arithmetic section to perform the actual computation on operands and a Control section that retrieves and decodes instructions from memory and directs the functional processes of the system. The CPU also contains an Operator's Console to allow manual control of the computer. The functional capability of the CPU can be expanded by the addition of the Program Restrict System, Stall Alarm, Interval Timer, Address Trap, and other options.

2-2 CENTRAL PROCESSING UNIT REGISTERS

The CPU contains 9 arithmetic and control registers and two pseudo registers. Eight optional registers are available that provide additional control and arithmetic functions.

2-2.1 Registers Available to the Programmer

The programmable and otherwise accessible registers of the CPU are illustrated on Figure 2-2 and described in the following paragraphs.

A. A Register

The 24-bit A register is the major CPU register. It contains complete arithmetic and shift capabilities and is used as the input/output communication register.

B. E Register

The E register is a 24-bit general register. It can also be used as an extension of the A register for additional shifting and arithmetic capability.

C. D Register

The D (Double) register is a 48-bit pseudo-register consisting of the 24-bit A and E registers. It provides double-precision capability. The A register contains the 24 least significant bits and the E extension contains the 24 most significant bits.

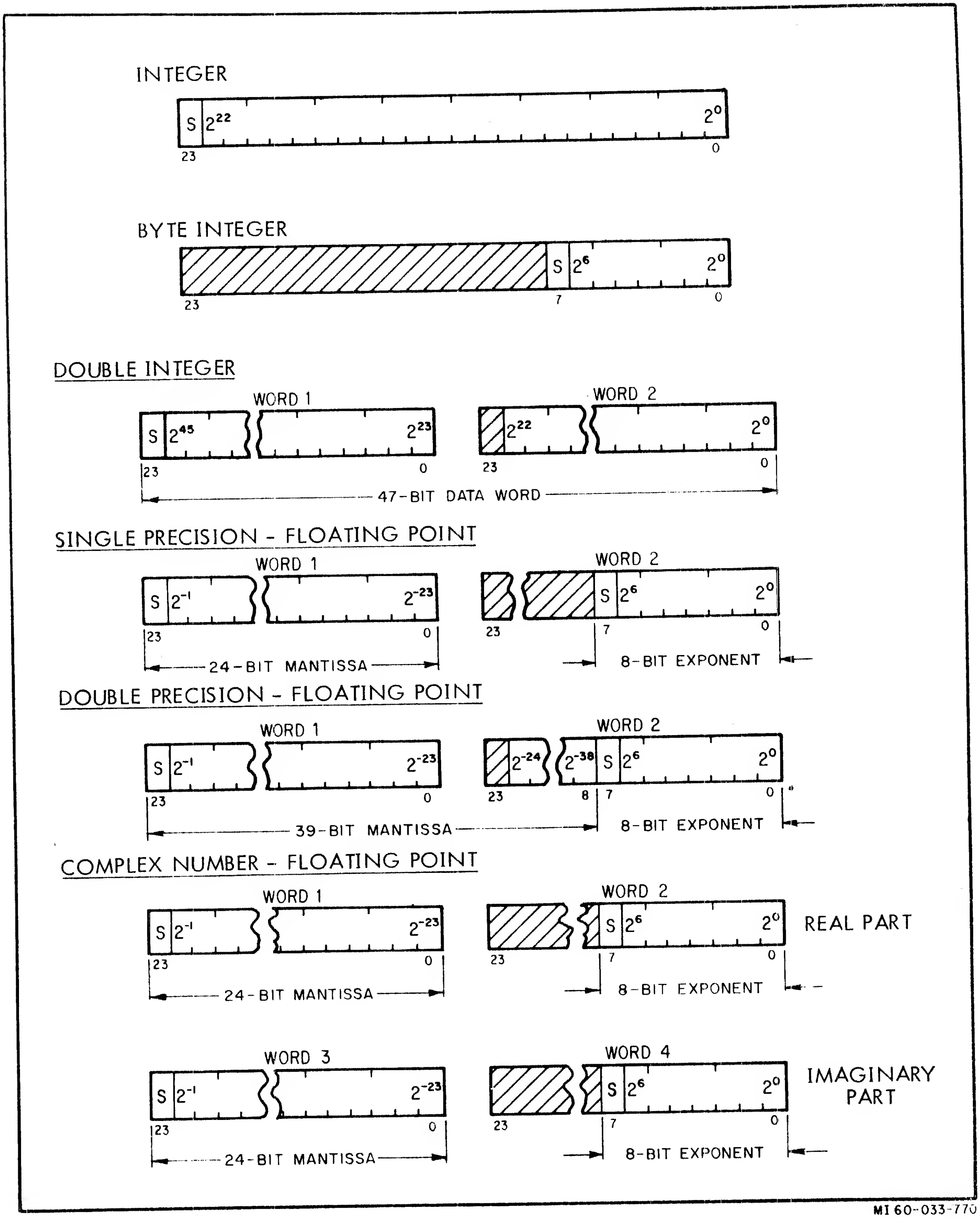


Figure 2-1. Data Word Formats

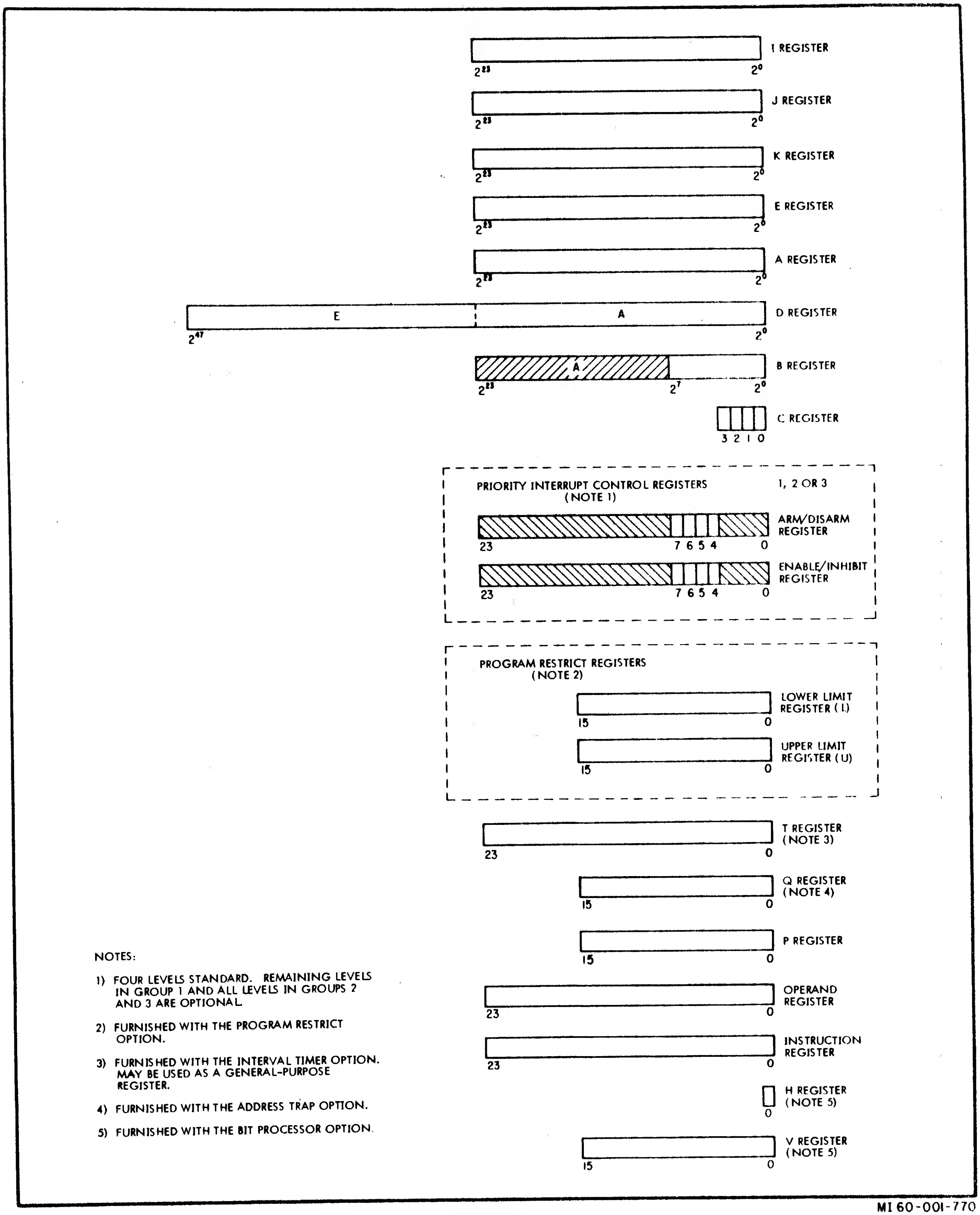


Figure 2-2. Central Processing Unit Registers

D. I, J, and K Registers

The I, J, and K registers are 24-bit general registers that may be used as index registers for address modification.

E. B Register

The B (Byte) register is a pseudo-register consisting of bits 0-7 of the A register. It provides byte manipulation capability.

F. C Register

The C (Condition) register is a 4-bit register that stores and displays the results of specific operations. The following conditions are displayed:

Bit 0 - indicates Overflow (logic 1) or No Overflow (logic 0)
Bit 1 - indicates Negative (logic 1) or Not Negative (logic 0)
Bit 2 - indicates Zero (logic 1) or Not Zero (Logic 0)
Bit 3 - indicates Positive (logic 1) or Not Positive (logic 0)

NOTE

Bits 1, 2, and 3 are mutually exclusive only when displaying the results of an operation.

2-2.2 Registers Not Available to the Programmer

Several registers within the CPU are not directly available to the programmer. These registers are described in the following paragraphs and illustrated in Figure 2-2.

A. P Register

The 16-bit P (Program address) register contains the address in memory from which the current instruction was taken for processing. A maximum of 65,536 memory locations can be accessed via the P register.

B. Operand Register

The 24-bit Operand register temporarily retains operands being transferred between maj sections of the CPU. The Operand register is not directly programmable, but functions as a console entry register.

C. Instruction Register

The 24-bit Instruction register contains instruction words after they have been accessed from memory and while they are being decoded. The instruction register is not programmable.

2-2.3 Optional Registers

Up to 8 additional registers can be added to the CPU with the incorporation of available options. These registers are described in the following paragraphs and are illustrated on Figure 2-2.

A. Priority Interrupt Control Registers

The Priority Interrupt Control registers are a pair of 24-bit registers that contain the Armed/Disarmed and Enabled/Disabled state of the discrete interrupt levels within a specified group (1, 2, or 3).

B. Program Restrict Registers

Two 16-bit registers are provided with the Program Restrict Option. These registers hold the upper (U register) and lower (L register) limits of an area in memory that is restricted from unauthorized access.

C. T Register

The T (timer) register is a 24-bit general register included with the Interval Timer option. This register may also be used as additional 24-bit general register that is accessed via the standard instruction set.

D. Q Register

The 16-bit Q (address Query) register is supplied with the Address Trap option. It stores a selected program address and when that address is reached in the program, a interrupt is generated.

E. H Register

A single-bit register supplied with the Bit Processor option. It allows bit manipulation.

F. V Register

The 16-bit V register is provided as part of the Bit Processor option. It stores a base address of an effective memory location containing the bit to be manipulated.

2-3 OPERATOR'S CONSOLE

The Series 6000 operator's console contains controls and indicators for:

a) starting and halting operations
b) entering data into the CPU registers and memory and for selecting the CPU registers for display and/or access.

The switches on the console provide a means for entering data into the CPU register and memory and for manually controlling CPU operations. Indicators display the contents of the CPU registers and memory and the status of the various operations. Figure 2-3 illustrates the controls and indicators located on the operator's console. Table 2-1 provides a functional description of the controls and indicators. The callouts reference the controls and indicators to the appropriate portion of Table 2-1.

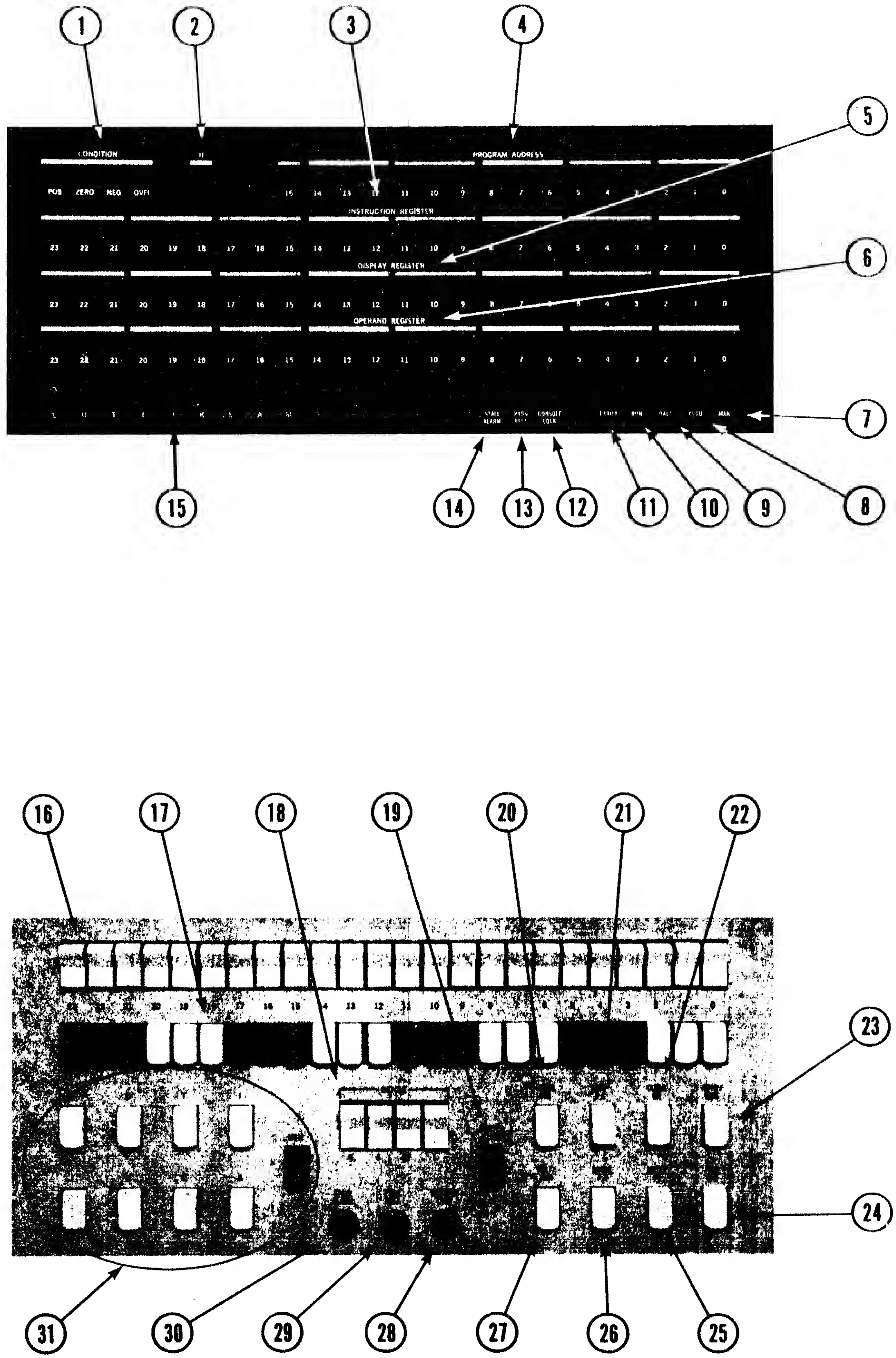


Figure 2-3. Series 6000 Console

Table 2-1. Controls and Indicators

REFERENCE (Figure 2-3)	CONTROL/ INDICATOR	FUNCTION
1	CONDITION Indicators	Displays the status of the CPU (i.e., the contents of the Condition register).
2	H Register Indicator	Displays the content of the H register.
3	INSTRUCTION REGISTER Indicators	Displays the contents of the INSTRUCTION REGISTER
4	PROGRAM ADDRESS Indicators	Displays the memory location from which the next instruction will be called for processing (i.e., the contents of the PROGRAM ADDRESS register).
5	DISPLAY REGISTER Indicators	Displays the contents of the register selected by the Register Select Switches. (See Item 31).
6	OPERAND REGISTER Indicators	Displays the contents of the OPERAND REGISTER.
7	MAN. Indicator	Indicates the CPU is in the MANual mode.
8	AUTO. Indicator	Indicates the CPU is in the AUTOmatic mode.
9	HALT Indicator	Indicates a CPU HALT condition.
10	RUN Indicator	Indicates the CPU is engaged in processing.
11	PARITY Indicator	Indicates the detection of a memory parity error.
12	CONSOLE LOCK Indicator	Indicates the CONSOLE LOCK has been enabled. (See Item 28).
13	PROG. REST. Indicator	Indicates the PROGram RESTrict system has been enabled. (See Item 29).
14	STALL ALARM Indicator	Indicates the STALL ALARM has been enabled. (See Item 30).
15	Selected Register Indicators	Indicates which CPU register has been selected (see Item 31) for display and/or entry.
16	Switch Register	Holds a 24-bit operand for subsequent transfer to one of the CPU registers upon execution of the TSI, TSJ, TSK, TSE, TSA or TST instruction.
17	Entry Switches	Enter ONEs into the corresponding bit positions of the OPERAND REGISTER.
18	Sense Switches 1, 2, 3, and 4	May be set and queried under program control.
19	MASTER CLEAR	Clears all CPU registers and initializes all control logic.

REFERENCE (Figure 2-3)	CONTROL/ INDICATOR	FUNCTION
20	ENTER REG	AUTO mode - replaces the contents of the selected register or memory location with the contents of the OPERAND REGISTER and then clears the OPERAND REGISTER. If MEM is selected, the PROGRAM ADDRESS is automatically advanced after the OPERAND REGISTER is cleared and the contents of the memory location defined by the new PROGRAM ADDRESS is displayed. MAN mode - replaces the contents of the selected register or memory location with the contents of the OPERAND REGISTER.
21	ADV PA	Advances the PROGRAM ADDRESS by one.
22	CLEAR OR	Clears the OPERAND REGISTER.
23	AUTO/MAN	Sets the operating mode to AUTOmatic or MANual. (See ENTER REG).
24	EXU IR	Executes the instruction in the INSTRUCTION REGISTER, advances the PROGRAM ADDRESS and loads the next instruction into the INSTRUCTION REGISTER.
25	ENTER IR	Replaces the contents of the INSTRUCTION REGISTER with the contents of the OPERAND REGISTER.
26	ENTER PA	Replaces the contents of the PROGRAM ADDRESS register with the contents of bits 0-16 of the OPERAND REGISTER, accesses the instruction at the new PROGRAM ADDRESS and loads the INSTRUCTION REGISTER and OPERAND REGISTER accordingly.
27	HALT/RUN	HALT - stops all processing and activates all console controls (unless disabled by the CONSOLE LOCK). RUN - starts program execution and deactivates all console controls except MASTER CLEAR.
28	CONSOLE LOCK Key Switch	Enables the CONSOLE LOCK and deactivates all switches.
29	PROG. REST. Key Switch	Enables the PROGram RESTrict system.
30	STALL ALARM Key Switch	Enables the STALL ALARM.
31	Register Select Switches	Each switch selects a specific register for display and/or entry and lights the appropriate register indicator (Item 15). Only one register may be selected at any one time. L - Lower Limit register U - Upper Limit register T - T register I - I register J - J register K - K register E - E register A - A register MEM - Memory location specified by the PROGRAM ADDRESS.

2-4 PRIORITY INTERRUPT SYSTEM

The priority interrupt system provides added program control of input/output operations and computations. Recognition of special external conditions is performed on the basis of predetermined priority. The priority interrupt system consists of four separate interrupt groups, 0 through 3. Group 0 contains eight optional executive traps. Groups 1 through 3 consists of 24 interrupt levels per group. Each computer is supplied with four interrupt levels. The DC 6024/1 has 68 additional interrupt levels available as options. Twenty additional interrupt levels are available as options for the DC 6024/3.

2-4.1 Group 0 (Executive Traps)

The optional executive traps are eight hard-wired, armed and enabled, interrupt levels. Once triggered by its external source, the executive trap becomes active immediately, i.e., with no higher priority interrupt active. Executive trap assignments are as follows:

Level 0	-	Power Down	Power Fail-Safe/Restore
Level 1	-	Power Up	
Level 2	-	Program Restrict	
Level 3	-	Instruction Trap	
Level 4	-	Stall Alarm	
Level 5	-	Interval Timer	
Level 6	-	Not Assigned	
Level 7	-	Address Trap	

2-4.2 Group 1, 2, and 3 (External Interrupts)

Since the need for and application of interrupts are so varied, maximum flexibility is obtained by providing a number of interrupts that can be program-controlled and are not permanently assigned. Groups 1, 2 and 3 provide a total of 72 such interrupts for the DC 6024/1. The DC 6024/3 uses only Group 1 to provide a total of 24 interrupts.

Four levels (8 through 11) in Group 1 are standard in all DC 6024 systems. The remaining levels in Group 1 and all levels in Groups 2 and 3 are optional in units of 4 levels.

Two 24-bit registers are associated with each external interrupt group (1-3). The first of these is the Arm/Disarm register. This register contains the armed/disarmed status of each interrupt level in the group. The second register - the Enable/Inhibit register - contains the enabled/inhibited status of each interrupt level within the group.

Interrupt levels are triggered by an external source. The contents of the Arm/Disarm and Enable/Inhibit registers determine whether the triggered interrupt will be ignored, held, or allowed to become active. A given interrupt level must be both armed and enabled before it becomes active. If the interrupt level is disarmed, the external trigger will be ignored. If an interrupt level is armed but inhibited (i.e., not enabled) it is held in a "waiting" state until such time as it is enabled under program control. Figure 2-4 illustrates the significance of the Arm/Disarm and Enable/Inhibit registers in this operation.

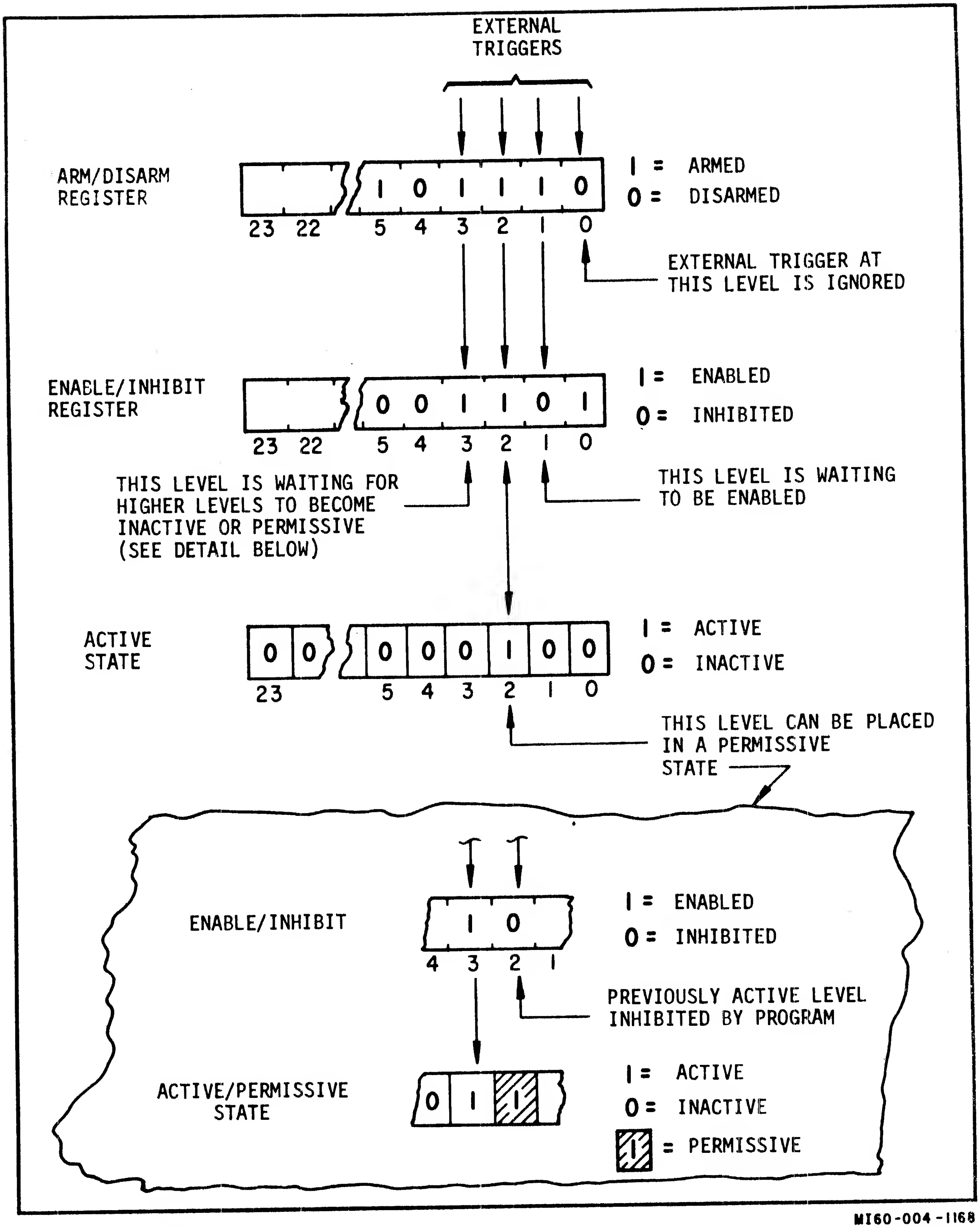


Figure 2-4. DC 6024 External Interrupt Control

Once an interrupt request (i.e., external trigger) passes both the Arm/Disarm and Enable/Inhibit tests, it becomes active as soon as the instruction being executed is completed assuming no higher priority level is active and that external interrupts are not being held by an HXI instruction.

After an interrupt becomes active, it is possible to inhibit that level under program control and permit an armed and enabled lower priority level to become active. If, for example, active level 2 (in Figure 2-4) is inhibited by the program, level 3 becomes active immediately. When the inhibited level (2) is re-enabled, it is restored to its normal priority. This "permissive" state of the interrupts permits on-line modification of the interrupt priority structure.

Program control of external interrupts is afforded by several instructions in the Series 6000 repertoire. Individual levels can be selectively (unitarily) armed, disarmed, enabled or inhibited or an entire group of interrupts can be simultaneously controlled. The HXI and RXI (Hold and Release eXternal Interrupts) instructions can prohibit and restore the activation of any external interrupt. The HXI instruction will positively prohibit the activation of any external interrupt (other than the levels currently active) regardless of that interrupt's armed/enabled status. This prohibition would insure that an interrupt processing routine would not be interrupted. This "hold" can only be released by an RXI instruction. For a detailed description of all priority interrupt instructions, refer to the appropriate portion of Section III.

2-4.3 Interrupt Processing Considerations

Each interrupt level has a memory location assigned for its exclusive use. This applies to both the Executive Traps (Group 0) and the external interrupts (Groups 1-3). Table 2-2 lists the dedicated memory locations for the interrupt system.

Table 2-2. Dedicated Memory Locations, Series 6000 Priority Interrupts

ADDRESSES (OCTAL)	FUNCTION
60-67	EXECUTIVE TRAPS (GROUP 0)
70-117	EXTERNAL INTERRUPTS (GROUP 1)
120-147	EXTERNAL INTERRUPTS (GROUP 2)
150-177	EXTERNAL INTERRUPTS (GROUP 3)

Table 2-3 lists the group and level of interrupt, its associated octal address, and its suggested assignment. For example, Group 1, level 8 is octal address 100 and is assigned to the input of the teletypewriter. The executive trap assignments are predetermined and are not reassignable.

When an interrupt level becomes active, the dedicated memory location is accessed. The instruction stored in that location is executed as though it were the next sequential location.

Although any instruction may be stored in the dedicated location, the Branch and Save return-Long (BSL) instruction provides a means for returning to the point of interrupt plus restoring the original machine status (i.e., the contents of the Condition register). Thus, the BSL instruction normally provides the entry for the interrupt processing routine.

Table 2-3. Interrupt Assignments

Group, Level	Address (Octal)	Function	
0,0	60	Power Down	Executive Traps
0,1	61	Power Up	Executive Traps
0,2	62	Program Restrict	Executive Traps
0,3	63	Instruction Trap	Executive Traps
0,4	64	Stall Alarm	Executive Traps
0,5	65	Interval Timer	Executive Traps
0,6	66	Not Assigned	Executive Traps
0,7	67	Address Trap	Executive Traps
1,0-7	70-77	Not Assigned	
1,8	100	Console Typewriter (Input)	
1,9	101	Console Typewriter (Output)	
1,10	102	Card Reader (Input)	
1,11	103	Card Punch (Output)	
1,12	104	Tape Reader (Input)	
1,13	105	Tape Punch (Output)	
1,14	106	Disc Status	
1,15	107	Magnetic Tape Status	
1,16-23	110-117	Not Assigned	
2,0-23	120-147	Not Assigned	
3,0-23	150-177	Not Assigned	

A means of exit from the interrupt routine is the Branch and Reset interrupt-Long (BRL) instruction. Normally, the BRL instruction would make use of an indirect reference (*) to the address previously referenced by the BSL instruction upon entering the routine. If this is done, the Condition register is restored to its original contents (at the time the interrupt occurred).

The BRL instruction resets the highest active trap or external interrupt level provided that external interrupts are not being "held" (HXI instruction). Active traps are always reset by the BRL.

2-5 PROGRAM RESTRICT/INSTRUCTION TRAP OPTION

The optional Program Restrict system allows areas of memory to be defined under program control and protected from unauthorized access. The Instruction Trap, furnished with the Program Restrict system, provides a means for preventing the execution of certain instructions.

2-5.1 Functional Description

The CPU operates with the Program Restrict system enabled or disabled depending on the position of the PROGRAM RESTRICT key switch. When the restrict system is disabled, all memory is accessible.

When the restrict system is enabled, the computer operates in three distinct modes as established under program control. The three Program Restrict modes are defined below:

(a) Unrestricted (Mode 0) - Programs working in this mode may access and alter any location in memory.

(b) Restricted/Privileged (Mode 1) - Programs operating in this mode may access and load from any location in memory; however, Mode 1 programs may not alter the contents of, or transfer control to, any memory location outside the specified limits.

(c) Restrict/Unprivileged (Mode 2) - Programs operating in this mode may not reference, in any manner, any memory location outside the specified limits.

Once established, the restrict mode is maintained by two flags operating concurrently. The mode flags can be set to one of three significant states to establish Mode 0, 1 or 2.

Control over the restrict system is maintained by the Program Restricted Flag (PRF). The PRF operates under the conditions and in the manner outlined below.

(a) The PRF may be set only when the restrict system is enabled (i.e., when the PROGRAM RESTRICT key switch is on).

(b) When in Mode 1 or 2, the PRF is set when an instruction transfers control into the restricted area of memory.

(c) When MASTER CLEAR is depressed, the PRF is set and Mode 2 is established automatically. MASTER CLEAR also clears the limit registers. Once a violation is made, the only way of recovering manually is by disabling the PROGRAM RESTRICT mode switch with the key and then depressing the MASTER CLEAR switch.

(d) Priority Interrupts reset (turn off) the PRF, rendering the mode flags ineffective until control is returned to the restricted area and the PRF is set again.

(e) When the PRF is set, executive trap interrupt 02 (group 0 level 2) occurs if a restrict violation takes place, except when operating in the HALT mode. (A restrict violation consists of any attempt to violate the conditions established by the modes.) The one exception to this is the Branch and Link Unrestricted (BLU) instruction. The BLU instruction has been implemented as an executive call to allow restricted programs to communicate directly with the resident operating system without being trapped. The BLU instruction resets the PRF and transfers control unconditionally to the address specified by the instruction.

(f) When the restrict system is enabled and in Mode 1 or 2, the console HALT switch can be activated only if the PRF is set. (This prohibits halting the machine during an interrupt routine.) If the CPU is operating in Mode 0, the only method of halting operation is to disable the restrict system (turn the key switch OFF) and activate the HALT switch.

(g) If the CPU is being controlled manually (HALT mode) and the restrict switch is in Mode 1 or 2 (PRF set), violations will be treated as follows:

(1) If the violating instruction is a BRANCH or an attempt to transfer an operand into memory, it will be treated as a No OPeration (NOP) instruction.

(2) If the violation attempts to transfer an operand from memory, all ZEROs will be retrieved and loaded into the destination register. (Mode 2 only.)

(3) If the normal advance of the PROGRAM ADDRESS causes the violation, all ZEROs will be loaded into the OPERAND REGISTER. The INSTRUCTION REGISTER will remain the same.

(4) If an attempt is made to enter an address into the PROGRAM ADDRESS REGISTER that is outside the bounds specified by the LIMIT REGISTERS, the PROGRAM ADDRESS REGISTER will not be loaded, and ZEROs will be retrieved from memory.

2-5.2 Program Control

The restricted area of memory is defined by two special registers, the Lower Limit (LL) and Upper Limit (UL) registers. Each register retains a 16-bit address that defines one limit of the restricted area.

Two instructions, Transfer Double to Limit registers (TDL) and Transfer Limit registers to Double (TLD), are provided for operating the Limit registers. The limits are defined by executing a TDL instruction where D = E and A, bits E_{15}-E_0 specify the lower limit and A_{15}-A_0 specify the upper limit. The TDL instruction also establishes the restrict mode by setting the mode flags with bits A_{21} and A_{22}. The bit configuration determines which mode will be established as shown in Table 2-4.

Table 2-4. PRF Bit Configuration

Bit A_{22}	Bit A_{21}	Mode
0	0	0
0	1	1
1	0	2
1	1	0

NOTE

If an attempt is made to execute the TDL instruction while in Mode 1 or 2, the Instruction Trap is activated.

The TLD instruction provides a method for saving the contents of the limit registers plus the status of the mode flags. The contents of the LL register is transferred to bits E_{15}-E_0 and the contents of the UL register is transferred to A_{15}-A_0. The mode flag bit configuration is retained in bits A_{22} and A_{21}.

2-5.3 Instruction Trap

The Instruction Trap is enabled and disabled by the PROGRAM RESTRICT key switch. When the PRF is OFF, the Instruction Trap is inhibited.

If an attempt is made to execute any of the instructions listed below, an executive trap interrupt 03 occurs at group 0 level 03. The interrupt routine may then examine the trapped instruction and determine what action is to be taken. The affected instructions are:

(a) Halt (HLT)
(b) Output Data Word (ODW)
(c) Input Data Word (IDW)
(d) Output Command Word (OCW)
(e) Input Status Word (ISW)
(f) Output Address Word (OAW)
(g) Input Address Word (IAW)
(h) Hold eXternal Interrupts (HXI)
(i) Release eXternal Interrupts (RXI)
(j) Unitarily Arm group 1 interrupts (UA1)
(k) Unitarily Arm group 2 interrupts (UA2)
(l) Unitarily Arm group 3 interrupts (UA3)
(m) Unitarily Disarm group 1 interrupts (UD1)
(n) Unitarily Disarm group 2 interrupts (UD2)
(o) Unitarily Disarm group 3 interrupts (UD3)
(p) Unitarily Enable group 1 interrupts (UE1)
(q) Unitarily Enable group 2 interrupts (UE2)
(r) Unitarily Enable group 3 interrupts (UE3)
(s) Unitarily Inhibit group 1 interrupts (UI1)
(t) Unitarily Inhibit group 2 interrupts (UI2)
(u) Unitarily Inhibit group 3 interrupts (UI3)
(v) Transfer Double to group 1 (TD1)
(w) Transfer Double to group 2 (TD2)
(x) Transfer Double to group 3 (TD3)
(y) Transfer Double to Limit registers (TDL)

2-6 STALL ALARM OPTION

The optional stall alarm detects a hangup or stall condition in the CPU by monitoring certain functions in the computer. If the function or operation is not completed before a predetermined time elapses, an executive trap interrupt is generated.

2-6.1 Functional Description

The stall alarm is enabled or disabled by the STALL ALARM key switch. Operation of the CPU is not affected when the stall alarm is disabled; however, a stall condition will not be recognized.

Execution of certain instructions or certain conditions existing within the CPU starts a binary counter in the stall alarm logic to monitor the operation. If after 128 machine cycles the operation is not complete or the conditions have not been corrected, an interrupt is generated on executive trap level 04.

The stall alarm executive trap causes the CPU to save the present contents of the variou registers so as not to lose data. After preserving the register contents, the CPU halts.

2-6.2 Program Control

There are no special programming considerations for the stall alarm option.

2-7 INTERVAL TIMER OPTION

The interval timer option consists of a 24-bit register (T register) and the associated control logic. The timer measures elapsed CPU time or real time as desired. A time interval may be preset in the interval timer under program control. When not employed as an interval timer, the T register can be used as a general purpose register.

2-7.1 Functional Description

The interval timer is decremented every ten machine cycles in all timing operations. The timer operates in one of two modes (processor time or clock time) depending on which timir signal is used to strobe the timer (T register).

In the processor time mode, the CPU's "T" timing pulses are used as the clock for the interval timer. This mode measures elapsed time of selected computations or operations within the CPU. When this mode is selected, the interval timer operates only during the time the CPU is running.

The clock time mode measures real time by counting down the time preset in the T register. A selected time interval is preset in the T register under program control and the time is counted down by using the CPU's "C" timing pulses as a strobe. In this mode, the interval timer is operable anytime power is applied to the computer.

When used as an interval timer, an executive trap interrupt is generated when the preset time is counted down to zero. The executive trap used is interrupt group 0, level 5.

2-7.2 Program Control

Interval timer operation is controlled with three instructions; Hold Interval Timer (HIT) Release Processor Time (RPT), or Release Clock Time (RCT). The HIT instruction inhibits interval timer operation in that a timing operation is prevented from starting or is halted while in progre The timer remains halted until it is released by a RPT or RCT instruction. The RPT instruction enables the interval timer in the processor time mode to allow the timer to measure elapsed CPL time. Time used for Automatic Block Controller operation is not measured as CPU time. Clock time mode operation is enabled with the RCT instruction. The interval timer counts CPU "C" timing pulses continuously.

Executive trap 05 interrupt is triggered when the contents of the T register are decremented to zero. Generation of the interrupt does not inhibit the timer.

When used as a general purpose register, the T register recognizes the various arithmet and transfer instructions associated with T register operation.

2-8 ADDRESS TRAP OPTION

The address trap option queries each referenced memory address and compares it to the address preset in the 16-bit Query register. Coincidence between the two addresses causes the executive trap interrupt to be generated.

2-8.1 Functional Description

The Query register is loaded with a 16-bit address. This address corresponds to a memory location that possesses some significance to the program. Each time this memory address is referenced, executive trap interrupt 07 is generated to inform the computer.

Operation of the address trap can be enabled or disabled with bit 23 of the 24-bit memory word used to load the Query register.

If bit 23 is set (ONE), the address trap is disabled. The address trap is enabled when bit 23 is reset or ZERO. Disabling the trap inhibits the executive trap interrupt.

2-8.2 Program Control

Control of the address trap is with the Transfer Memory to Query register (TMQ) instruction. This instruction transfers the selected memory location from memory via the Operand register to the Query register. This location contains a 16-bit address word and the enable/disable control bit (bit 23). With the Query register loaded and the address trap enabled, an interrupt is generated each time a reference is made to the memory location corresponding to the address stored in the Query register. If it is desired that a reference to the selected memory location be recognized only once, a second TMQ instruction should be executed following the first interrupt with bit 23 set to ONE. This disables the address trap.

2-9 BIT PROCESSOR OPTION

The bit processor option consists of the single-bit H register, a 16-bit V register (Base register), and the associated control logic. The bit processor provides the capability to selectively change a bit in memory, store a bit in memory, or test a bit.

2-9.1 Functional Description

The 16-bit V register is loaded with a base address which specifies a memory location to be manipulated. This is accomplished by transferring a 16-bit memory address from the K register. The instruction word further defines the memory location, the specific bit, and the operation to be performed.

After the operation is performed on the selected bit, the results are displayed in the Condition register. Refer to Section III, paragraph 3-16.

2-9.2 Program Control

Two instruction formats are associated with bit processor operations. The first specifies a displacement (bits 0-7 of the instruction word) from the base address to specify the location to used. Five bits in the instruction word (bits 8-12) select the specific bit to be processed. The remaining (bits 13-24) contain the OP CODE which specifies the operation to be performed.

The second format is not used for specifying a bit in memory but for bit movement or transfer operations where a specific bit is not required. Bits 12-23 contain a 12-bit OP CODE and bits 0-11 are undefined.

A complete list of the instructions associated with bit processor operation is provided in Section III, paragraph 3-16. A brief description of each instruction is also provided.

2-10 HARDWARE BOOTSTRAP OPTION

Initially, all programs are stored in memory by a bootstrap loader program. The bootstr automatically loads a record(s) from the selected peripheral device. This record(s) normally con a more complex loader which allows other programs, operating systems, or other data to be storec in selected memory locations.

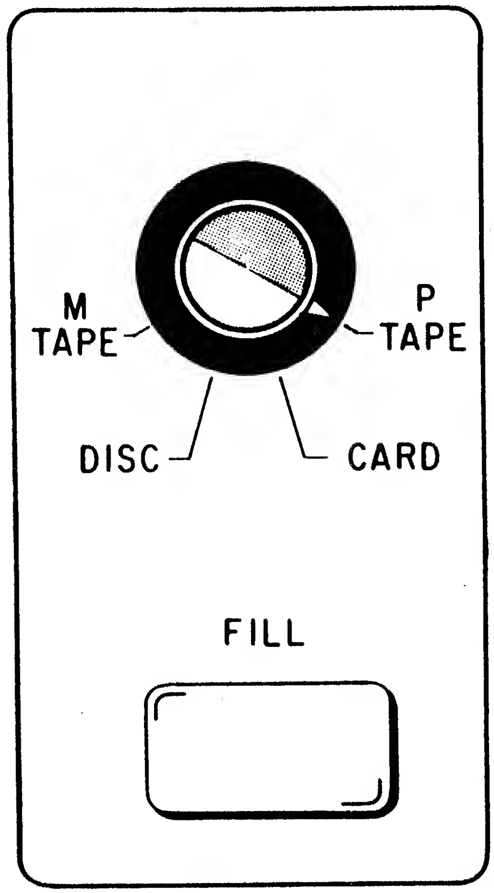
The optional hardware bootstrap automatically loads the appropriate bootstrap program into memory. At present, two bootstrap programs may be selected. They are the Paper Tape Reader or Disc bootstraps.

2-10.1 Paper Tape Reader Bootstrap Functional Description

The paper tape reader bootstrap option loads a program in memory that permits the absolute tape loader to be stored in memory from the paper tape reader. This bootstrap is selected by setting the Bootstrap Select switch (Figure 2-5) on the operator's console to P TAPE. Depressing the FILL (Figure 2-5) switch automatically stores and then executes the program provided in Table 2-5. The memory locations correspond to decoded counts of a binary counter that controls the bootstrap operation. The octal memory location is shown in parenthesis.

2-10.2 Disc Bootstrap Functional Description

The disc bootstrap option loads a program in memory that permits the link loader to be stored in memory from the disc. The disc bootstrap is selected by setting the Bootstrap Select switch (Figure 2-5) to DISC. Depressing the FILL switch (Figure 2-5) automatically stores and then executes the program provided in Table 2-6. The memory locations shown correspond to decoded counts of a binary counter that controls the bootstrap operation. The octal memory location is shown in parenthesis. The program generated by the bootstrap is provided in Table 2



MI60-034-770

Figure 2-5. Bootstrap Control Panel

Table 2-5. Hardware Bootstrap Program, Paper Tape Reader

Memory Location	Instruction Code	Operation	Operand
1 (01)	00030110	TOB	'110
2 (02)	00700100	OCW	CU
3 (03)	00720100	IDW	CU
4 (04)	00140000	COB	0
5 (05)	22200003	BOZ	*-2
6 (06)	63200004	TNJ	4
7 (07)	00420006	LLA	6
8 (10)	00724100	IDW*	CU
9 (11)	22600010	BNZ	*-1
10 (12)	23200007	BWJ	*-3
11 (13)	00240020	CZA	
12 (14)	22200020	BOZ	'20
13 (15)	15100020	TAM	'20, I
14 (16)	23100006	BWI	*-8

NOTE

1. SENSE switch 1 must be set when using the ASR-33 Paper Tape Reader.

2. CU = '100 for High-Speed Reader
   CU = '000 for ASR

Table 2-6. Hardware Bootstrap Program, Disc

Memory Location	Instruction Code	Operation	Operand
1 (01)	62500013	TOA	WC
2 (02)	00714500	OAW	CU
3 (03)	05000012	TMA	CW
4 (04)	00700500	OCW	CU
5 (05)	00730500	ISW	CU
6 (06)	22600005	BNZ	*-1
7 (07)	00110200	QBB	B7
8 (10)	22600005	BNZ	*-3
9 (11)	21000020	BUC	'20
10 (12)	40000000	DATA (CW)	B23
11 (13)	00000100	DAC (WC)	'100
12 (14)	00000020	DAC (BA)	'20

2-11 POWER FAILURE SHUTDOWN/RESTART OPTION

This option saves the operating program in the event of a power failure and the subsequent restart when power levels return to normal. The power failure shutdown/restart circuits inhibit the start of a new memory cycle during the power interruption.

2-11.1 Functional Description

The power failure shutdown/restore option monitors the input AC power source for the computer power supplies for fluctuations in power level. A decrease in the AC voltage level of approximately 10 percent causes an executive trap 00 interrupt (group 0, level 0) to be generated. A power down halt signal is generated and inserted in bit position 20 of the memory word that is prepared as a result of the interrupt. This notifies the program if the CPU was running or halted at the time of the power failure. If bit 20 is a logic ONE, the CPU was running. Bit 20 is a logic ZERO if the CPU was halted.

A master clear signal is generated 100 microseconds after the executive trap is generated to complete the shutdown process. This 100 microsecond period allows the program time to save the data being processed.

When the AC power level returns to 95 percent of its nominal level, another master clear signal is generated to begin the restore process. The restart signal is generated approximately 120 milliseconds later. The restart signal generates an interrupt on executive trap 01 (group 0, level 1).

2-11.2 Program Control

There are no special programming considerations for the power failure shutdown/restore option.

INSTRUCTION SET

SECTION III
INSTRUCTION SET

3-1 INTRODUCTION

The Series 6000 instruction set consists of several functional groups or families of instructions. Among these are: Arithmetic; Branch; Compare; Input/Output; Logical; Shift; Transfer; etc. Each group, in turn, is composed of individual instructions that perform specific functions.

Through the application of the instruction set, the programmer has access to each memory location and major register in the Series 6000 CPU. In addition, the instruction set provides for the alteration and control of program flow, manipulation and modification (arithmetic and logical) of data, servicing of priority interrupts and control of Input/Output operations.

3-2 INSTRUCTION FORMATS

Each instruction is decoded from a 24-bit memory word. The instruction word bits define the operation to be performed and the manner in which it is to be performed. All instruction formats contain an operation code (OP CODE) that defines the general process that is to be undertaken (Add, Subtract, Interchange, etc.). The OP CODE may contain either six or twelve bits.

Additional bits in the instruction word specify how the general operation is to be performed. For example, when adding the contents of one register to the contents of another, the additional bits indicate which registers are involved.

Some instructions access memory and use formats that specify an address. The address bits are sometimes supplemented by special bits (indirect, index) in the instruction word. In other cases, the additional bits are not used for address modification, but are used to define a condition under which the specified memory location will be accessed or to indicate which of the CPU registers will be used in the operation. The appropriate formats are provided with the individual instruction descriptions.

3-3 INSTRUCTION FORMULA

The instruction formula, presented with each instruction description, provides a graphic representation of a 24-bit instruction word. The formula expresses an instruction word as a concatenation of its various fields where each field is represented by one or more octal digits. For example, the formula 21.*+X:a expresses a memory reference branch where: 21 represents a 6-bit (2 octal digits) OP CODE, * and X are additive quantities defining the indirect (*) and index (X) field, and "a" is a memory reference in a 15-bit address field.

The period (.) and colon (:) provide field separation in the formula, with the colon indicating right/left justification. All digits or references to the left of the colon are left-justified, and those to the right are right-justified in their respective fields. The absence of a colon indicates that all digits or references are left-justified in their fields. Examples of instruction formulas are shown on the following page.

```
          0034.
OP CODE ---/      \----------- BLANK FIELD (bits insignificant)

          0027. R. R
OP CODE ---/    |  \---------- REGISTER SPECIFICATION FIELD
                |
          REGISTER
          SPECIFICATION
          FIELD

          64. R:o
OP CODE ---/   | \------------ OPERAND FIELD
               |
          REGISTER
          SPECIFICATION
          FIELD

          0070. *+C. U
OP CODE ---/      |    \------ UNIT SPECIFICATION FIELD
                  |
          CHANNEL
          SPECIFICATION
          FIELD
          W/OVERRIDE
```

3-4 ADDRESSING TECHNIQUE

When memory exceeds 32,768 words, the Series 6000 automatic memory mapping scheme comes into consideration. Memory from 0 to 32,768 words comprises Map 0, while Map 1 consists of memory from 32,768 to 65,536 words.

The addressing technique considers the most significant bit of the Program Address register (P_{15}) as the Map bit. $P_{15} = 0$ specifies that the current map is Map 0, and $P_{15} = 1$ indicates Map 1. By performing a logical OR function between the Map bit and the immediate address reference, a program may directly address up to 32,768 words in its own map.

In Map 0 ($P_{15} = 0$), immediate address references may be indexed to access up to 65,536 words since ORing the Map bit would not alter the effective address. However, when in Map 1, Map 0 may not be referenced in this manner since all immediate references are biased by 100000_8.

NOTE

It should be noted that when the last location in Map 0 is used for an address reference, the Program Counter will have advanced by the time the effective address is computed and the address reference in this location will be biased by 100000_8 which places the address in Map 1.

Indirect references are not affected by the Map bit. This allows multilevel indirect references, indexed at any level, to access up to 65,536 words irrespective of the current map.

A group of "Long Branch" instructions have a 16-bit immediate address field, allowing Map 0 or Map 1 programs to directly address 0 - 65,536 words. The Long Branch instructions are not affected by the Map Bit.

The basic memory reference formats and their instruction formulas are illustrated in Figure 3-1. Figure 3-2 illustrates the memory referencing logic.

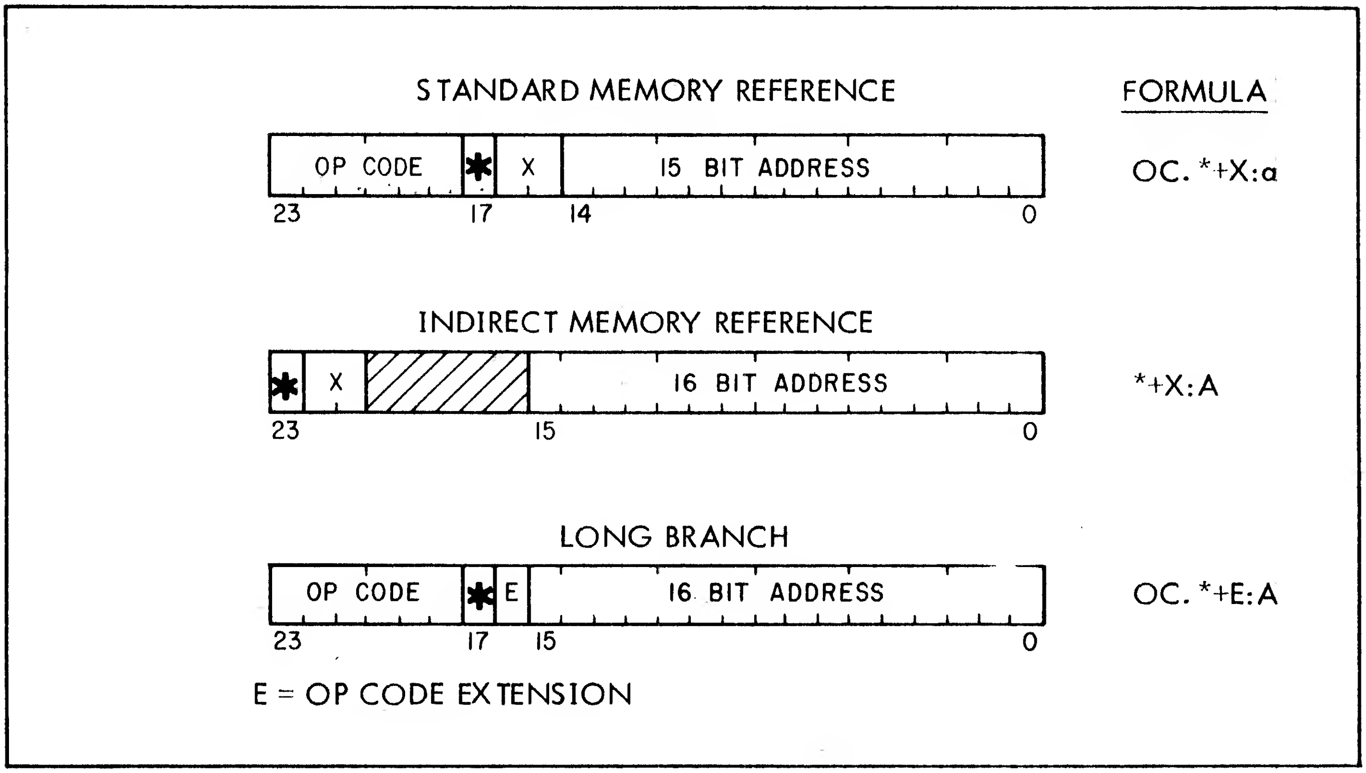
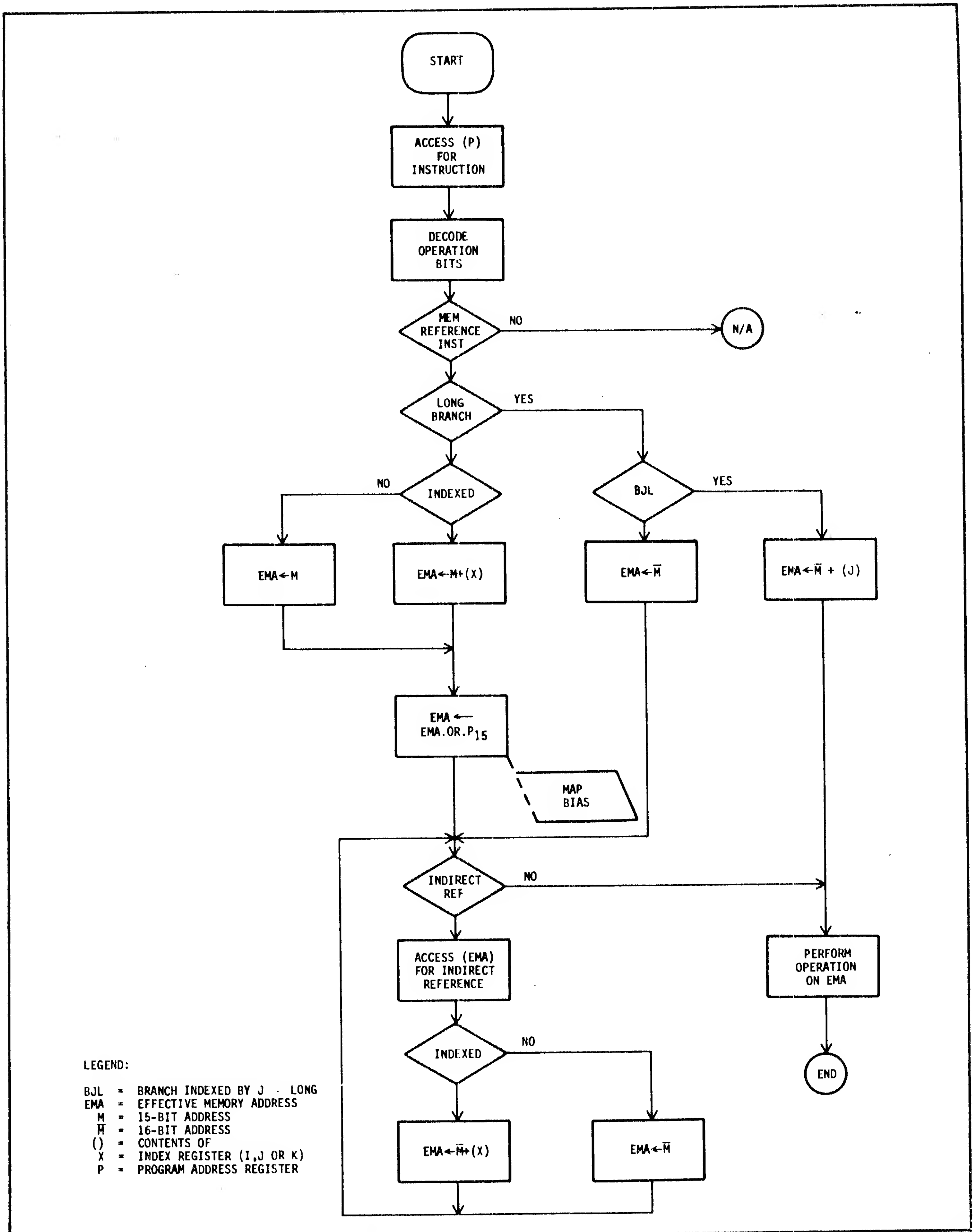


Figure 3-1. Series 6000 Memory Reference Formats and Formulas

3-5 INSTRUCTION DESCRIPTIONS

The following paragraphs describe, in detail, the various instructions in the Series 6000 repertoire. The instructions have been arranged by functional groups (Arithmetic, Branch, Compare, etc.). General information pertaining to the entire group is presented in the introductory paragraphs.



MI 60-003-1068

Figure 3-2. Memory Reference Logic

3-6 ARITHMETIC INSTRUCTIONS

The Series 6000 Arithmetic instruction group includes the standard arithmetic operations — addition, subtraction, multiplication and division — as well as square root, normalization and sign extension instructions. Also included are several register-to-register operations which compute the absolute value, negate or round off the contents, or negate the sign of one register and subsequently transfer its contents to a second register.

The arithmetic instruction mnemonics provide a brief definition of specific operations to be performed. The first letter of the mnemonic signifies the action or type of operation to be performed, the second letter identifies the first quantity or reference (R_1) to be used in the operation, and the third letter identifies the second reference (R_2). For example:

A M E

Add (Action to be performed)

Memory (R_1)

Register E (R_2)

In the majority of arithmetic instructions, the result of the operation remains in R_2 leaving R_1 unchanged (except where R_1 and R_2 are the same). Certain instructions — notably, those performing multiplication, division, sign extension and square root computation — do not comply with the R_1/R_2 conventions stated above. These instances are described thoroughly in the individual instruction descriptions.

Unless noted otherwise, each arithmetic operation causes the Condition (C) register to be set reflecting the status of the result. The various arithmetic conditions are defined below:

(a) Positive — Result is arithmetically greater than zero, indicated by a ONE in bit position 3 of the C register. A ZERO in bit position 3 indicates "Not Positive".

(b) Zero — All bits of the quantity under consideration are ZEROs, indicated by a ONE in bit position 2 of the C register. A ZERO in bit position 2 indicates "Not Zero".

(c) Negative — Result is arithmetically less than zero, indicated by a ONE in bit position 1 of the C register. A ZERO in bit position 1 indicates "Not Negative".

(d) Overflow — An Overflow results from an operation instead of displaying the status of an operand. As a general rule, an arithmetic Overflow will occur when a bit is carried into the designated sign bit position and not carried out or vice versa. An Overflow condition is indicated by a ONE in bit position 0 of the C register. A ZERO in bit position 0 indicates "No Overflow".

The following instructions are included in the Arithmetic Group.

MNEMONIC	INSTRUCTION	PAGE
AAA	Add A to A	3-16
AAE	Add A to E	3-16
AAI	Add A to I	3-16
AAJ	Add A to J	3-16
AAK	Add A to K	3-16
AAM	Add A to Memory	3-14
AAT	Add A to T	3-16
AEA	Add E to A	3-16
AEE	Add E to E	3-16
AEI	Add E to I	3-16
AEJ	Add E to J	3-16
AEK	Add E to K	3-16
AEM	Add E to Memory	3-14
AET	Add E to T	3-16
AIA	Add I to A	3-16
AIE	Add I to E	3-16
AII	Add I to I	3-16
AIJ	Add I to J	3-16
AIK	Add I to K	3-16
AIM	Add I to Memory	3-13
AIT	Add I to T	3-16
AJA	Add J to A	3-16
AJE	Add J to E	3-16
AJI	Add J to I	3-16
AJJ	Add J to J	3-16
AJK	Add J to K	3-16
AJM	Add J to Memory	3-13
AJT	Add J to T	3-16
AKA	Add K to A	3-16
AKE	Add K to E	3-16
AKI	Add K to I	3-16
AKJ	Add K to J	3-16
AKK	Add K to K	3-16
AKM	Add K to Memory	3-13
AKT	Add K to T	3-16
AMA	Add Memory to A	3-12
AMB	Add Memory to Byte	3-13
AMD	Add Memory to Double	3-12
AME	Add Memory to E	3-12
AMI	Add Memory to I	3-11
AMJ	Add Memory to J	3-11
AMK	Add Memory to K	3-11
AOA	Add Operand to A	3-14
AOB	Add Operand to Byte	3-15
AOE	Add Operand to E	3-14
AOI	Add Operand to I	3-14
AOJ	Add Operand to J	3-14
AOK	Add Operand to K	3-14
AOM	Add Operand to Memory	3-15

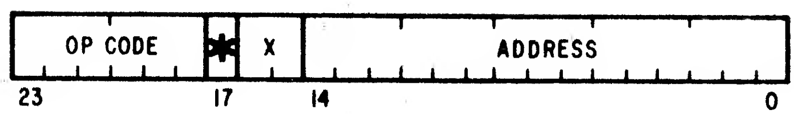
MNEMONIC	INSTRUCTION	PAGE
AOT	Add Operand to T	3-14
ATA	Add T to A	3-16
ATE	Add T to E	3-16
ATI	Add T to I	3-16
ATJ	Add T to J	3-16
ATK	Add T to K	3-16
ATT	Add T to T	3-16
AUM	Add Unity to Memory	3-11
DVI	DiVide by I	3-19
DVJ	DiVide by J	3-19
DVK	DiVide by K	3-19
DVM	DiVide by Memory	3-17
DVO	DiVide by Operand	3-18
DVT	DiVide by T	3-19
ESA	Extend Sign of A	3-20
ESB	Extend Sign of Byte	3-20
FNO	Floating NOrmalize	3-21
MYA	MultiplY by A	3-23
MYE	MultiplY by E	3-23
MYI	MultiplY by I	3-23
MYJ	MultiplY by J	3-23
MYK	MultiplY by K	3-23
MYM	MultiplY by Memory	3-22
MYO	MultiplY by Operand	3-22
MYT	MultiplY by T	3-23
NAA	Negate of A to A	3-24
NAE	Negate of A to E	3-24
NAI	Negate of A to I	3-24
NAJ	Negate of A to J	3-24
NAK	Negate of A to K	3-24
NAT	Negate of A to T	3-24
NBB	Negate of Byte to Byte	3-23
NDD	Negate of Double to Double	3-25
NEA	Negate of E to A	3-24
NEE	Negate of E to E	3-24
NEI	Negate of E to I	3-24
NEJ	Negate of E to J	3-24
NEK	Negate of E to K	3-24
NET	Negate of E to T	3-24
NIA	Negate of I to A	3-24
NIE	Negate of I to E	3-24
NII	Negate of I to I	3-24
NIJ	Negate of I to J	3-24
NIK	Negate of I to K	3-24
NIT	Negate of I to T	3-24
NJA	Negate of J to A	3-24
NJE	Negate of J to E	3-24
NJI	Negate of J to I	3-24
NJJ	Negate of J to J	3-24
NJK	Negate of J to K	3-24
NJT	Negate of J to T	3-24

MNEMONIC	INSTRUCTION	PAGE
NKA	Negate of K to A	3-24
NKE	Negate of K to E	3-24
NKI	Negate of K to I	3-24
NKJ	Negate of K to J	3-24
NKK	Negate of K to K	3-24
NKT	Negate of K to T	3-24
NSA	Negate Sign of A	3-25
NSE	Negate Sign of E	3-25
NSI	Negate Sign of I	3-25
NSJ	Negate Sign of J	3-25
NSK	Negate Sign of K	3-25
NST	Negate Sign of T	3-25
NTA	Negate of T to A	3-24
NTE	Negate of T to E	3-24
NTI	Negate of T to I	3-24
NTJ	Negate of T to J	3-24
NTK	Negate of T to K	3-24
NTT	Negate of T to T	3-24
PAA	Positive of A to A	3-27
PAE	Positive of A to E	3-27
PAI	Positive of A to I	3-27
PAJ	Positive of A to J	3-27
PAK	Positive of A to K	3-27
PAT	Positive of A to T	3-27
PBB	Positive of Byte to Byte	3-26
PDD	Positive of Double to Double	3-26
PEA	Positive of E to A	3-27
PEE	Positive of E to E	3-27
PEI	Positive of E to I	3-27
PEJ	Positive of E to J	3-27
PEK	Positive of E to K	3-27
PET	Positive of E to T	3-27
PIA	Positive of I to A	3-27
PIE	Positive of I to E	3-27
PII	Positive of I to I	3-27
PIJ	Positive of I to J	3-27
PIK	Positive of I to K	3-27
PIT	Positive of I to T	3-27
PJA	Positive of J to A	3-27
PJE	Positive of J to E	3-27
PJI	Positive of J to I	3-27
PJJ	Positive of J to J	3-27
PJK	Positive of J to K	3-27
PJT	Positive of J to T	3-27
PKA	Positive of K to A	3-27
PKE	Positive of K to E	3-27
PKI	Positive of K to I	3-27
PKJ	Positive of K to J	3-27
PKK	Positive of K to K	3-27
PKT	Positive of K to T	3-27
PTA	Positive of T to A	3-27

MNEMONIC	INSTRUCTION	PAGE
PTE	Positive of T to E	3-27
PTI	Positive of T to I	3-27
PTJ	Positive of T to J	3-27
PTK	Positive of T to K	3-27
PTT	Positive of T to T	3-27
REA	Round of E to A	3-28
REE	Round of E to E	3-28
REI	Round of E to I	3-28
REJ	Round of E to J	3-28
REK	Round of E to K	3-28
RET	Round of E to T	3-28
RIA	Round of I to A	3-28
RIE	Round of I to E	3-28
RII	Round of I to I	3-28
RIJ	Round of I to J	3-28
RIK	Round of I to K	3-28
RIT	Round of I to T	3-28
RJA	Round of J to A	3-28
RJE	Round of J to E	3-28
RJI	Round of J to I	3-28
RJJ	Round of J to J	3-28
RJK	Round of J to K	3-28
RJT	Round of J to T	3-28
RKA	Round of K to A	3-28
RKE	Round of K to E	3-28
RKI	Round of K to I	3-28
RKJ	Round of K to J	3-28
RKK	Round of K to K	3-28
RKT	Round of K to T	3-28
RTA	Round of T to A	3-28
RTE	Round of T to E	3-28
RTI	Round of T to I	3-28
RTJ	Round of T to J	3-28
RTK	Round of T to K	3-28
RTT	Round of T to T	3-28
SAE	Subtract A from E	3-32
SAI	Subtract A from I	3-32
SAJ	Subtract A from J	3-32
SAK	Subtract A from K	3-32
SAT	Subtract A from T	3-32
SEA	Subtract E from A	3-32
SEI	Subtract E from I	3-32
SEJ	Subtract E from J	3-32
SEK	Subtract E from K	3-32
SET	Subtract E from T	3-32
SIA	Subtract I from A	3-32
SIE	Subtract I from E	3-32
SIJ	Subtract I from J	3-32
SIK	Subtract I from K	3-32
SIT	Subtract I from T	3-32
SJA	Subtract J from A	3-32

MNEMONIC	INSTRUCTION	PAGE
SJE	Subtract J from E	3-32
SJI	Subtract J from I	3-32
SJK	Subtract J from K	3-32
SJT	Subtract J from T	3-32
SKA	Subtract K from A	3-32
SKE	Subtract K from E	3-32
SKI	Subtract K from I	3-32
SKJ	Subtract K from J	3-32
SKT	Subtract K from T	3-32
SMA	Subtract Memory from A	3-29
SMB	Subtract Memory from Byte	3-30
SMD	Subtract Memory from Double	3-30
SME	Subtract Memory from E	3-29
SMI	Subtract Memory from I	3-29
SMJ	Subtract Memory from J	3-29
SMK	Subtract Memory from K	3-29
SOA	Subtract Operand from A	3-31
SOB	Subtract Operand from Byte	3-31
SOE	Subtract Operand from E	3-31
SOI	Subtract Operand from I	3-31
SOJ	Subtract Operand from J	3-31
SOK	Subtract Operand from K	3-31
SOT	Subtract Operand from T	3-31
SRE	Square Root - Extended	3-33
SRT	Square Root	3-33
STA	Subtract T from A	3-32
STE	Subtract T from E	3-32
STI	Subtract T from I	3-32
STJ	Subtract T from J	3-32
STK	Subtract T from K	3-32

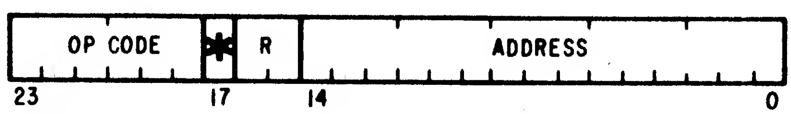
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
30. *+X:a	Add Unity to Memory	M,C	AUM



The contents of the effective memory address are incremented by one.

Cycles .... 3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
41. *+1:a	Add Memory to I	I,C	AMI
41. *+2:a	J	J,C	AMJ
41. *+3:a	K	K,C	AMK



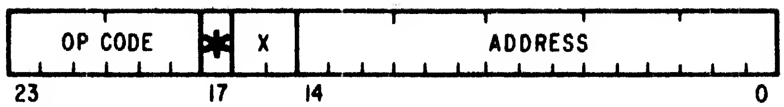
The contents of the effective memory address are algebraically added to the contents of register I, J or K.

NOTE

The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

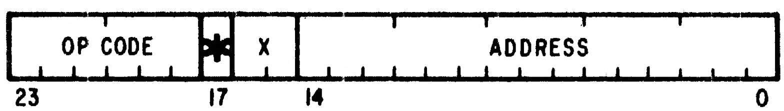
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
42. *+X:a	Add Memory to E	E,C	AME
43. *+X:a	A	A,C	AMA



The contents of the effective memory address are algebraically added to the contents of register E or A.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

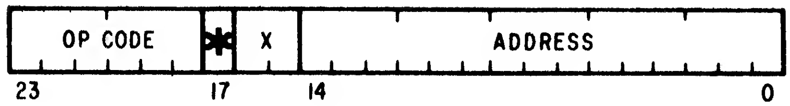
44. *+X:a	Add Memory to Double	E,A,C	AMD



The contents of the effective memory address (EMA) and the next sequential memory address (EMA +1) are algebraically added to the contents of register D according to the double integer format defined in Section II.

Cycles ....3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

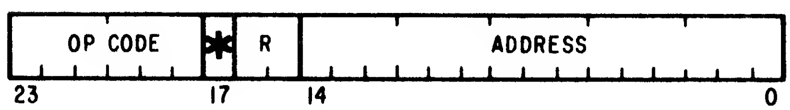
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
45. *+X:a	Add Memory to Byte	A,C	AMB



Bits 0-7 of the contents of the effective memory address are algebraically added to the contents of register B (A_0-A_7). Bits 8-23 of register A are unchanged.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

46. *+1:a	Add I to Memory	M,C	AIM
46. *+2:a	J	M,C	AJM
46. *+3:a	K	M,C	AKM



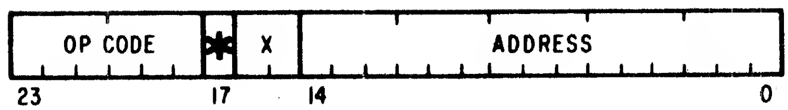
The 24-bit contents of register I, J or K are algebraically added to the contents of the effective memory address.

NOTE

The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles ....3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

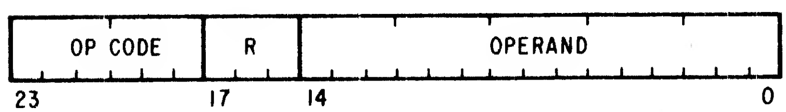
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
47. *+X:a	Add E to Memory	M,C	AEM
50. *+X:a	A	M,C	AAM



The contents of register E or A are algebraically added to the contents of the effective memory address.

Cycles ....3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

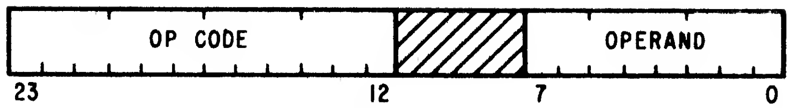
64. 1:o	Add Operand to I	I,C	AOI
64. 2:o	J	J,C	AOJ
64. 3:o	K	K,C	AOK
64. 4:o	E	E,C	AOE
64. 5:o	A	A,C	AOA
64. 6:o	T	T,C	AOT



The 15-bit unsigned operand is algebraically added to the contents of the specified register.

Cycles 1
Reference pages 3-1, 3-5

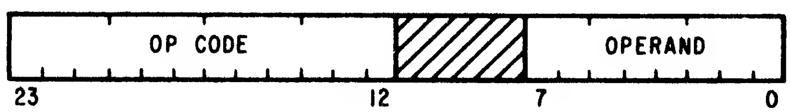
INSTRUCTION FORMULA		FUNCTION	REGISTERS AFFECTED	MNEMONIC
0012:o		Add Operand to Byte	A,C	AOB

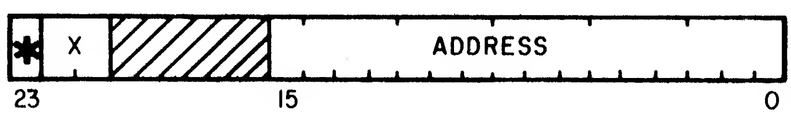


The 8-bit signed operand is algebraically added to the contents of the B register. (A_0-A_7). Bits 8-23 of register A are unchanged.

Cycles 1
References pages 3-1, 3-5

0074:o *+X:A	(word 1) (word 2)	Add Operand to Memory	M,C	AOM

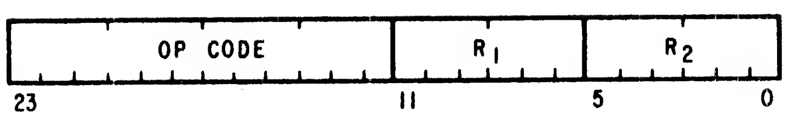




The 8-bit signed operand is algebraically added to the contents of the effective memory address.

Cycles .... 4 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0020.01.01	Add I to I	I,C	AII
0020.01.02	J	J,C	AIJ
0020.01.04	K	K,C	AIK
0020.01.10	E	E,C	AIE
0020.01.20	A	A,C	AIA
0020.01.40	T	T,C	AIT
0020.02.01	Add J to I	I,C	AJI
0020.02.02	J	J,C	AJJ
0020.02.04	K	K,C	AJK
0020.02.10	E	E,C	AJE
0020.02.20	A	A,C	AJA
0020.02.40	T	T,C	AJT
0020.04.01	Add K to I	I,C	AKI
0020.04.02	J	J,C	AKJ
0020.04.04	K	K,C	AKK
0020.04.10	E	E,C	AKE
0020.04.20	A	A,C	AKA
0020.04.40	T	T,C	AKT
0020.10.01	Add E to I	I,C	AEI
0020.10.02	J	J,C	AEJ
0020.10.04	K	K,C	AEK
0020.10.10	E	E,C	AEE
0020.10.20	A	A,C	AEA
0020.10.40	T	T,C	AET
0020.20.01	Add A to I	I,C	AAI
0020.20.02	J	J,C	AAJ
0020.20.04	K	K,C	AAK
0020.20.10	E	E,C	AAE
0020.20.20	A	A,C	AAA
0020.20.40	T	T,C	AAT
0020.40.01	Add T to I	I,C	ATI
0020.40.02	J	J,C	ATJ
0020.40.04	K	K,C	ATK
0020.40.10	E	E,C	ATE
0020.40.20	A	A,C	ATA
0020.40.40	T	T,C	ATT



The contents of R_1 are algebraically added to the contents of R_2.

Cycles 1
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
57. *+X:a	DiVide by Memory	E,A,C	DVM

OP CODE	*	X	ADDRESS
23	17	14	0

The double-precision contents of register D (E and A) are algebraically divided by the single-precision contents of the effective memory address. The signed, single-precision, quotient is left in A and the remainder is left in E. The remainder will have the same sign as the original dividend and the Condition register will be set according to the status of the quotient.

NOTES

(1) If it is desired to divide a single-precision number in A by memory, an Extend Sign of A (ESA) instruction should be executed prior to the DVM. This will establish the proper format for the dividend.

(2) If the contents of E are equal to, or greater then, the contents of memory, an Overflow condition will result and the Condition register will be set accordingly.

Cycles .... 15 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
61.0:o	DiVide by Operand	E,A,C	DVO

OP CODE	R	OPERAND
23	17	14 0

The double-precision contents of register D (E and A) are algebraically divided by the 15-bit unsigned operand. The signed, single-precision, quotient is left in A and the remainder is left in E. The remainder will have the same sign as the original dividend and the Condition register will be set according to the status of the quotient.

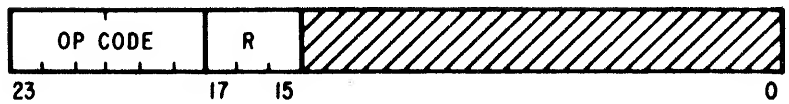
NOTES

(1) If it is desired to divide a single-precision number in A by the operand, an Extend Sign of A (ESA) instruction should be executed prior to the DVO. This will establish the proper format for the dividend.

(2) If the contents of E are equal to, or greater than, the operand, an Overflow condition will result and the Condition register will be set accordingly.

Cycles 15
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
61.1	DiVide by I	E,A,C	DVI
61.2	J	E,A,C	DVJ
61.3	K	E,A,C	DVK
61.6	T	E,A,C	DVT



The double-precision contents of register D (E and A) are algebraically divided by the contents of the specified register. The signed, single-precision, quotient is left in A and the remainder is left in E. The remainder will have the same sign as the original dividend and the Condition register will be set according to the status of the quotient.

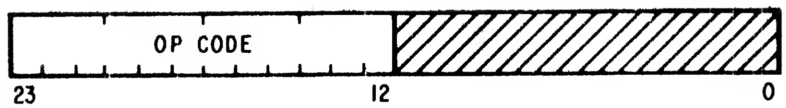
NOTES

(1) If it is desired to divide a single-precision number in A by by the contents of the specified register, an Extend Sign of A (ESA) instruction should be executed prior to the divide instruction. This will establish the proper format for the dividend.

(2) If the contents of E are equal to, or greater than, the contents of the specified register, an Overflow condition will result and the Condition register will be set accordingly.

Cycles 15
Reference pages 3-1, 3-5

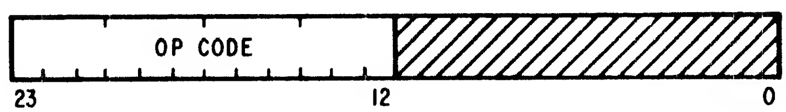
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0037.	Extend Sign of A	E,C,A	ESA



The state of the sign bit (A_{23}) of register A is copied into all 24 bit positions of register E and bit A_{23} is then set to zero. This forms a double-precision number in E and A.

Cycles 1
Reference pages 3-1, 3-5

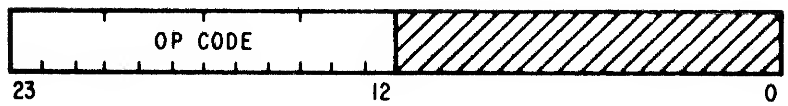
0010.	Extend Sign of Byte	A,C	ESB



The state of the register B sign bit (A_7) is copied into bit positions A_8-A_{23}, forming a sign extension of the byte.

Cycles 1
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0054.	Floating NOrmalize	E,A,I,C	FNO



The contents of register D (E and A) are shifted left arithmetically until bit E_{22} differs from E_{23}. The negative shift count (i.e., the number of shifts performed) replaces the contents of register I.

Example: Convert a double-precision integer in D to double-precision floating point format.

FNO	Normalize
TIB	Position exponent in byte. (A_0-A_7)
BOZ*+2	If result is zero, no exponent adjustment is necessary.
AOB 46	Adjust shift count.

NOTES

There are four special cases where the shifting process differs from that described above.

(1) If the binary pattern 11000...0 is detected in register D, normalization is terminated to avoid creating the invalid pattern 10000...0.

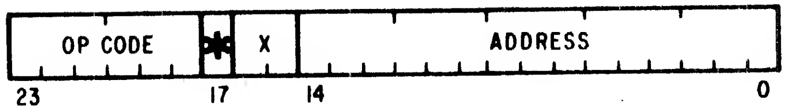
(2) If the invalid binary pattern 10000...0 is detected, it is shifted right one position, producing the pattern 11000...0. The shift count is adjusted accordingly.

(3) If the pattern 00000...0 is detected, the shift count is set to -177_8, making a zero less significant than any other value.

(4) If an Overflow condition is present when beginning the operation, the contents of register D are arithmetically shifted right one position. The shift count is set to ONE and the sign of D is complemented.

Cycles .... 2+[(n-1)/4]
Reference pages 3-1, 3-5, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
56. *+X:a	MultiplY by Memory	E, A, C	MYM



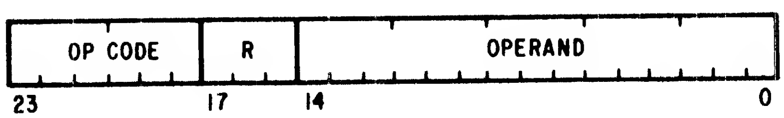
The contents of register A are algebraically multiplied by the contents of the effective memory address. The double-precision product replaces the previous contents of register D (E and A).

NOTE

An Overflow will result if the full-scale negative number (1000. . . . 00) is used as both the multiplier and multiplicand.

Cycles . . . . 8 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

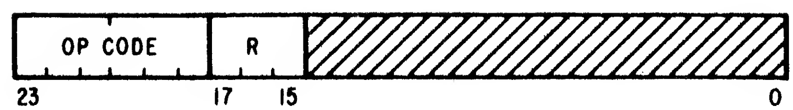
60. 0:o	MultiplY by Operand	E, A, C	MYO



The contents of register A are algebraically multiplied by the 15-bit unsigned operand in the instruction word. The double-precision product replaces the previous contents of register D (E and A).

Cycles 8
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
60. 1	MultiplY by I	E,A,C	MYI
60. 2	J	E,A,C	MYJ
60. 3	K	E,A,C	MYK
60. 4	E	E,A,C	MYE
60. 5	A	E,A, C	MYA
60. 6	T	E,A,C	MYT



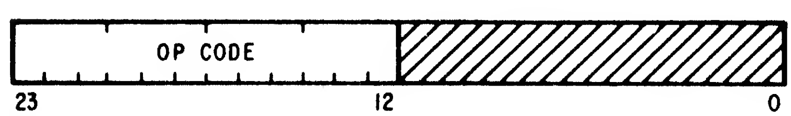
The contents of register A are algebraically multiplied by the contents of the specified register. The double-precision product replaces the previous contents of register D (E and A).

NOTE

An Overflow will result if the full-scale negative number (1000. . . . 00) is used as both the multiplier and multiplicand.

Cycles 8
Reference pages 3-1, 3-5

0005.	Negate of Byte to Byte	A,C	NBB



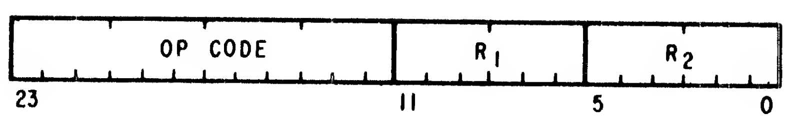
The contents of register B (A_0-A_7) are two's complemented. Bit positions A_8-A_{23} are unchanged.

NOTE

An Overflow will result when negating 2^7 (full-scale negative byte).

Cycles 1
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0022.01.01	Negate of I to I	I, C	NII
0022.01.02	J	J, C	NIJ
0022.01.04	K	K, C	NIK
0022.01.10	E	E, C	NIE
0022.01.20	A	A, C	NIA
0022.01.40	T	T, C	NIT
0022.02.01	Negate of J to I	I, C	NJI
0022.02.02	J	J, C	NJJ
0022.02.04	K	K, C	NJK
0022.02.10	E	E, C	NJE
0022.02.20	A	A, C	NJA
0022.02.40	T	T, C	NJT
0022.04.01	Negate of K to I	I, C	NKI
0022.04.02	J	J, C	NKJ
0022.04.04	K	K, C	NKK
0022.04.10	E	E, C	NKE
0022.04.20	A	A, C	NKA
0022.04.40	T	T, C	NKT
0022.10.01	Negate of E to I	I, C	NEI
0022.10.02	J	J, C	NEJ
0022.10.04	K	K, C	NEK
0022.10.10	E	E, C	NEE
0022.10.20	A	A, C	NEA
0022.10.40	T	T, C	NET
0022.20.01	Negate of A to I	I, C	NAI
0022.20.02	J	J, C	NAJ
0022.20.04	K	K, C	NAK
0022.20.10	E	E, C	NAE
0022.20.20	A	A, C	NAA
0022.20.40	T	T, C	NAT
0022.40.01	Negate of T to I	I, C	NTI
0022.40.02	J	J, C	NTJ
0022.40.04	K	K, C	NTK
0022.40.10	E	E, C	NTE
0022.40.20	A	A, C	NTA
0022.40.40	T	T, C	NTT



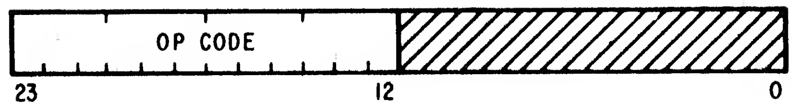
The two's complement of the contents of R_1 replace the previous contents of R_2.

NOTE

An Overflow will result when negating 2^{23} (full-scale negative number).

Cycles 1
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0033.	Negate of Double to Double	E,A,C	NDD



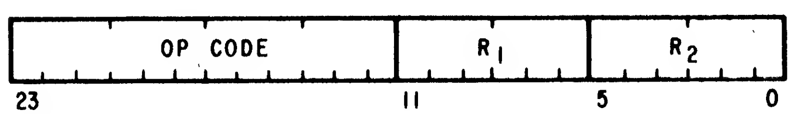
The contents of register D (E and A), in double-precision format, are two' s complemented.

NOTE

An Overflow will result when negating 2^{46} (full-scale negative double integer).

Cycles 2
Reference pages 3-1, 3-5

0032. 01. 01	Negate Sign of I	I,C	NSI
0032. 02. 02	J	J,C	NSJ
0032. 04. 04	K	K, C	NSK
0032. 10. 10	E	E,C	NSE
0032. 20. 20	A	A,C	NSA
0032.40.40	T	T,C	NST



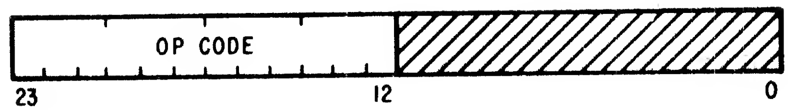
The sign bit of the specified register is complemented.

NOTE

An overflow will result when negating a full-scale negative.

Cycles 1
Reference pages 3-1, 3-5

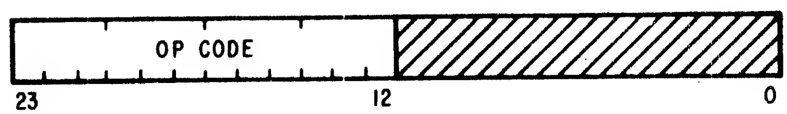
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0006.	Positive of Byte to Byte	A, C	PBB



The absolute value of the contents of register B (A_0-A_7) is placed in register B.

Cycles 1
Reference pages 3-1, 3-5

0034.	Positive of Double to Double	E, A, C	PDD



The absolute value of the contents of register D is placed in register D according to the double integer format defined in Section II.

Cycles 2
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0023.01.01	Positive of I to I	I,C	PII
0023.01.02	J	J,C	PIJ
0023.01.04	K	K,C	PIK
0023.01.10	E	E,C	PIE
0023.01.20	A	A,C	PIA
0023.01.40	T	T,C	PIT
0023.02.01	Positive of J to I	I,C	PJI
0023.02.02	J	J,C	PJJ
0023.02.04	K	K,C	PJK
0023.02.10	E	E,C	PJE
0023.02.20	A	A,C	PJA
0023.02.40	T	T,C	PJT
0023.04.01	Positive of K to I	I,C	PKI
0023.04.02	J	J,C	PKJ
0023.04.04	K	K,C	PKK
0023.04.10	E	E,C	PKE
0023.04.20	A	A,C	PKA
0023.04.40	T	T,C	PKT
0023.10.01	Positive of E to I	I,C	PEI
0023.10.02	J	J,C	PEJ
0023.10.04	K	K,C	PEK
0023.10.10	E	E,C	PEE
0023.10.20	A	A,C	PEA
0023.10.40	T	T,C	PET
0023.20.01	Positive of A to I	I,C	PAI
0023.20.02	J	J,C	PAJ
0023.20.04	K	K,C	PAK
0023.20.10	E	E,C	PAE
0023.20.20	A	A,C	PAA
0023.20.40	T	T,C	PAT
0023.40.01	Positive of T to I	I,C	PTI
0023.40.02	J	J,C	PTJ
0023.40.04	K	K,C	PTK
0023.40.10	E	E,C	PTE
0023.40.20	A	A,C	PTA
0023.40.40	T	T,C	PTT

OP CODE	R_1	R_2
23	11	5 0

The absolute value of the contents of R_1 replaces the previous contents of R_2.

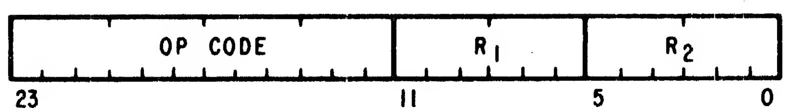
NOTE

An Overflow will result when negating a full-scale negative.

Cycles 1

Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0075.01.01	Round of I to I	I, C	RII
0075.01.02	J	J, C	RIJ
0075.01.04	K	K, C	RIK
0075.01.10	E	E, C	RIE
0075.01.20	A	A, C	RIA
0075.01.40	T	T, C	RIT
0075.02.01	Round of J to I	I, C	RJI
0075.02.02	J	J, C	RJJ
0075.02.04	K	K, C	RJK
0075.02.10	E	E, C	RJE
0075.02.20	A	A, C	RJA
0075.02.40	T	T, C	RJT
0075.04.01	Round of K to I	I, C	RKI
0075.04.02	J	J, C	RKJ
0075.04.04	K	K, C	RKK
0075.04.10	E	E, C	RKE
0075.04.20	A	A, C	RKA
0075.04.40	T	T, C	RKT
0075.10.01	Round of E to I	I, C	REI
0075.10.02	J	J, C	REJ
0075.10.04	K	K, C	REK
0075.10.10	E	E, C	REE
0075.10.20	A	A, C	REA
0075.10.40	T	T, C	RET
0075.40.01	Round of T to I	I, C	RTI
0075.40.02	J	J, C	RTJ
0075.40.04	K	K, C	RTK
0075.40.10	E	E, C	RTE
0075.40.20	A	A, C	RTA
0075.40.40	T	T, C	RTT

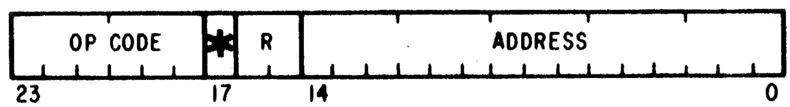


Round the contents of R_1 as a function of A and place the result in R_2.

If bit A_{22} is a ONE, the contents of R_1 +1 are transferred to R_2. If A_{22} is ZERO, the contents of R_1 replace the previous contents of R_2. In either case, R_1 is unchanged except when the same as R_2.

Cycles 1
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
51. *+1:a	Subtract Memory from I	I, C	SMI
51. *+2:a	J	J, C	SMJ
51. *+3:a	K	K, C	SMK



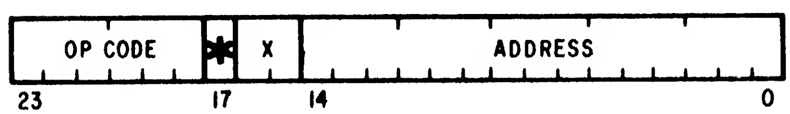
The contents of the effective memory address are algebraically subtracted from the contents of register I, J or K.

NOTE

The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

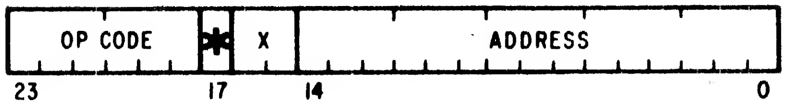
52. *+X:a	Subtract Memory from E	E, C	SME
53. *+X:a	A	A, C	SMA



The contents of the effective memory address are algebraically subtracted from the contents of register E or A.

Cycles .... 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
54. *+X:a	Subtract Memory from Double	E,A,C	SMD



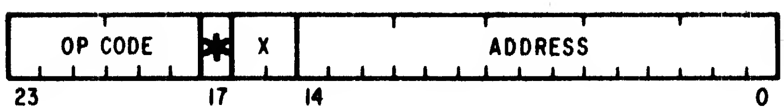
The contents of the effective memory address (EMA) and the next sequential address (EMA +1) are algebraically subtracted from the contents of register D (E and A), according to the double integer format defined in Section II.

NOTE

Failure to adhere to the double integer format will provide unpredictable results.

Cycles .... 3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

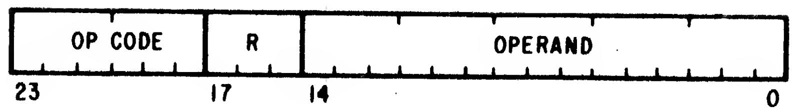
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
55. *+X:a	Subtract Memory from Byte	A,C	SMB



The contents of bits 0-7 of the effective memory address are algebraically subtracted from register B (A_0-A_7). Bits A_8-A_{23} are unaffected.

Cycles .... 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-5

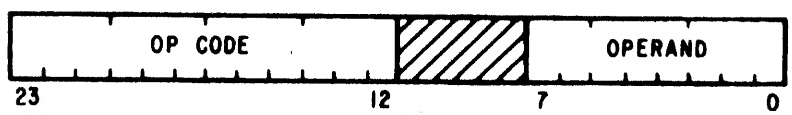
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
65. 1:o	Subtract Operand from I	I, C	SOI
65. 2:o	J	J, C	SOJ
65. 3:o	K	K, C	SOK
65. 4:o	E	E, C	SOE
65. 5:o	A	A, C	SOA
65. 6:o	T	T, C	SOT



The 15-bit unsigned operand is algebraically subtracted from the contents of the specified register.

Cycles 1
Reference pages 3-1, 3-5

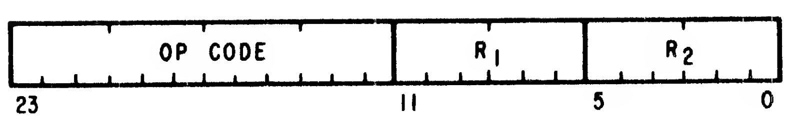
0013:o	Subtract Operand from Byte	A, C	SOB



The 8-bit signed operand is algebraically subtracted from the contents of register B (A_0-A_7). Bits A_8-A_{23} are unaffected.

Cycles 1
Reference pages 3-1, 3-5

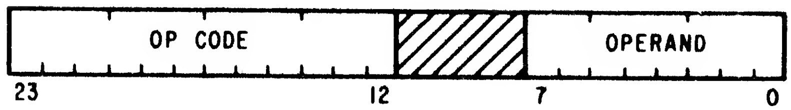
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0021.01.02	Subtract I from J	J,C	SIJ
0021.01.04	K	K,C	SIK
0021.01.10	E	E,C	SIE
0021.01.20	A	A,C	SIA
0021.01.40	T	T,C	SIT
0021.02.01	Subtract J from I	I,C	SJI
0021.02.04	K	K,C	SJK
0021.02.10	E	E,C	SJE
0021.02.20	A	A,C	SJA
0021.02.40	T	T,C	SJT
0021.04.01	Subtract K from I	I,C	SKI
0021.04.02	J	J,C	SKJ
0021.04.10	E	E,C	SKE
0021.04.20	A	A,C	SKA
0021.04.40	T	T,C	SKT
0021.10.01	Subtract E from I	I,C	SEI
0021.10.02	J	J,C	SEJ
0021.10.04	K	K,C	SEK
0021.10.20	A	A,C	SEA
0021.10.40	T	T,C	SET
0021.20.01	Subtract A from I	I,C	SAI
0021.20.02	J	J,C	SAJ
0021.20.04	K	K,C	SAK
0021.20.10	E	E,C	SAE
0021.20.40	T	T,C	SAT
0021.40.01	Subtract T from I	I,C	STI
0021.40.02	J	J,C	STJ
0021.40.04	K	K,C	STK
0021.40.10	E	E,C	STE
0021.40.20	A	A,C	STA



The contents of R_1 are algebraically subtracted from R_2.

Cycles 1
Reference pages 3-1, 3-5

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0076:014	Square Root	E,A,C	SRT
0076:027	Square Root — Extended	E,A,C	SRE



The contents of register A are treated as a 23-bit positive integer. The square root of this quantity is placed in register A, right justified, and the remainder is placed in register E so that:

$$\text{root}^2 + \text{remainder} = \text{original integer.}$$

If the sign bit (23) of register A is set, the Condition register will be set to OVERFLOW.

SRT generates a root of 12 significant bits; i.e., the true integer root of any positive integer in register A.

SRE generates a root of 23 significant bits. This extended significance is obtained by assuming 22 zeros to the right of bit A_0; effectively, multiplying the contents of A by 2^{22} and, consequently, the root by 2^{11}.

Consider the following examples where: A_n implies a binary point to the right of bit n.

Positive Integer	Instruction	Root (Octal)
2 at A_0	SRT	1 at A_0
2 at A_0	SRE	1.3240 at A_{11}
2 at A_{20}	SRT	1.3240 at A_{10}
2 at A_{20}	SRE	1.3240474 at A_{21}

Cycles: SRT 14
SRE 25
Reference pages 3-1, 3-5

This page left blank intentionally.

3-7 BRANCH INSTRUCTIONS

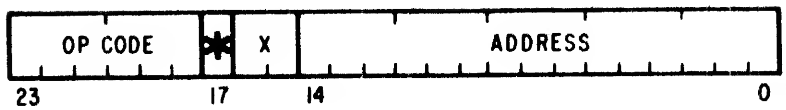
The Branch group of instructions can be divided into two basic types: conditional and unconditional branches. Conditional branches cause control to be transferred to a specified address upon detection of a certain machine condition as indicated by the contents of the Condition register. Unconditional branches cause control to be transferred unconditionally to a specified address.

The following instructions are included in the Branch group.

MNEMONIC	INSTRUCTION	PAGE
BJL*	Branch indexed by J — Long	3-38
BLI	Branch and Link I	3-39
BLJ	Branch and Link J	3-39
BLK	Branch and Link K	3-39
BLL*	Branch and Link (J) — Long	3-39
BLU	Branch and Link — Unrestricted	3-42
BNN	Branch on Not Negative	3-37
BNO	Branch on No Overflow	3-37
BNP	Branch on Not Positive	3-37
BNZ	Branch on Not Zero	3-37
BON	Branch On Negative	3-37
BOO	Branch On Overflow	3-37
BOP	Branch On Positive	3-37
BOZ	Branch On Zero	3-37
BRL*	Branch and Reset interrupts — Long	3-41
BSL*	Branch and Save return — Long	3-40
BUC	Branch UnConditionally	3-36
BUL*	Branch Unconditionally — Long	3-36
BWI	Branch When $I+1 \neq 0$	3-38
BWJ	Branch When $J+1 \neq 0$	3-38
BWK	Branch When $K+1 \neq 0$	3-38

*Long branches disregard the automatic memory mapping logic.

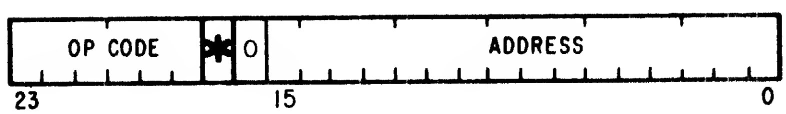
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
21. *+X:a	Branch UnConditionally	P	BUC



The contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address.

Cycles .... 1 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-35

26. *+0:A	Branch Unconditionally — Long	P	BUL



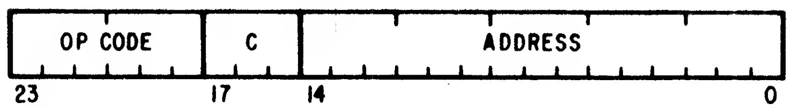
The contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address.

NOTE

> The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 1 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-35

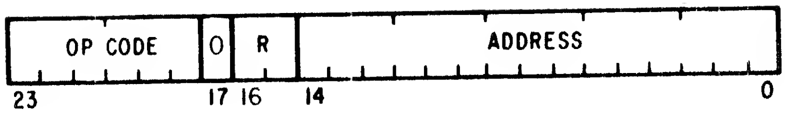
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
22.0:a	Branch On Overflow	P	BOO
22.1:a	On Negative	P	BON
22.2:a	On Zero	P	BOZ
22.3:a	On Positive	P	BOP
22.4:a	on No Overflow	P	BNO
22.5:a	on Not Negative	P	BNN
22.6:a	on Not Zero	P	BNZ
22.7:a	on Not Positive	P	BNP



The contents of the Condition register are tested for the specified condition. If the condition is present, the contents of the P register (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address. If the specified condition is not present, the program advances to the next sequential location (PROGRAM ADDRESS +1).

Cycles 1
Reference pages 3-1, 3-2, 3-35

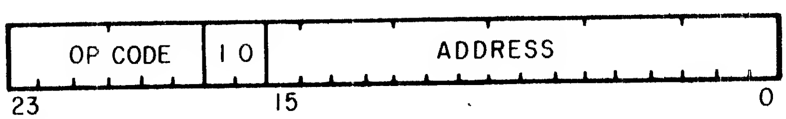
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
23.1:a	Branch When I+1 $\neq$ 0	I, P	BWI
23.2:a	J+1 $\neq$ 0	J, P	BWJ
23.3:a	K+1 $\neq$ 0	K, P	BWK



The contents of the specified register are incremented by one and then tested for zero. If the contents are not zero, the contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address. If the contents are zero, the program advances to the next sequential location (PROGRAM ADDRESS +1).

Cycles 1
Reference pages 3-1, 3-2, 3-35

23.4:A	Branch indexed by J — Long	P	BJL



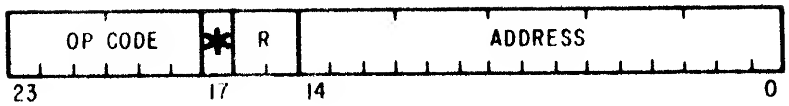
The contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address.

NOTE

The immediate memory reference is automatically indexed by J.

Cycles 1
Reference pages 3-1, 3-2, 3-35

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
24. *+1:a	Branch and Link I	I, P	BLI
24. *+2:a	J	J, P	BLJ
24. *+3:a	K	K, P	BLK



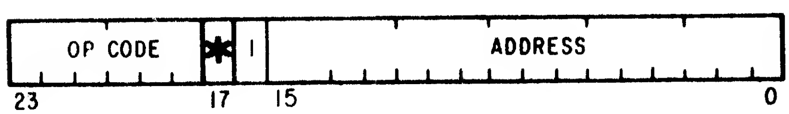
The contents of register I, J or K are replaced by the next sequential address (PROGRAM ADDRESS +1) and the contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address.

NOTE

> The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 1 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-35

26. *+2:A	Branch and Link (J) — Long	J, P	BLL



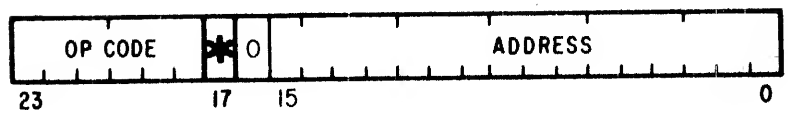
The contents of register J are replaced by the next sequential address (PROGRAM ADDRESS +1) and the contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address.

NOTE

> The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 1 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-35

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
25. *+0:A	Branch and Save return — Long	P	BSL



The next sequential address (PROGRAM ADDRESS +1), along with the contents of the Condition register are stored in the 16-bit effective memory address (EMA). The contents of register P (current PROGRAM ADDRESS) are then replaced by the address following the effective memory address (EMA +1).

This instruction is normally used to enter an interrupt subroutine because it provides a means of returning to the main program at the point of interrupt and saves the machine status (Condition) at the time of the interrupt.

NOTES

(1) The contents of the Condition register are stored in bit positions 16-19 of the EMA and the return address (PROGRAM ADDRESS +1) is stored in bits 0-15. The remaining bits are set to ZEROs; however, refer to note 2 for variation on bit 20.

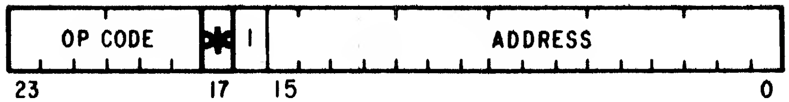
(2) If a "Power Down" interrupt occurs when the CPU is halted, bit 20 is set to ZERO. If the CPU is running when the "Power Down" interrupt occurs, bit 20 is set to ONE.

(3) The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

(4) External interrupts are prohibited for the period of one instruction following the execution of this instruction.

Cycles .... 2 (+1 per indirect reference)
Reference pages 2-11, 3-1, 3-2, 3-35

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
25. *+2:A	Branch and Reset interrupt — Long	C,P	BRL



The highest-level active interrupt is reset (i.e., returned to the inactive state) and the contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address.

BRL is normally used to exit an interrupt subroutine. If BRL contains an indirect reference, the last word in the indirect address chain contains the previous status (i.e., C register contents at the time of the interrupt) in bit positions M_{16}–M_{19} and the return address in bit positions M_0–M_{15} as a result of the BSL instruction. The C register is restored and the program branches to the return address (restarting the machine to the preinterrupt status).

Example:

```
        :
        TMA
L       AMA
        SMA          Interrupt occurs (EXM K).
        :
K       BSL    M     Dedicated interrupt location.

M       ***          M becomes L+1 as a result of BSL at K.
        :            The C register contents are stored in
        :            M16-M19.
        BRL*   M     Restore C register and return to L+1.
```

NOTES

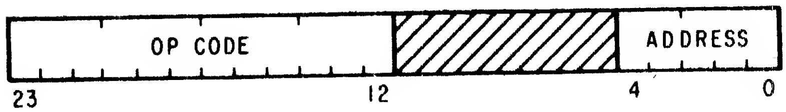
(1) The BRL will not reset the interrupt if external interrupts have been held by an HXI instruction. Control will be returned to the effective memory address.

(2) Executive traps, which are not affected by the HXI instruction, will be reset by the BRL.

(3) The immediate memory reference cannot be indexed; however, indexing indirect references is permitted.

Cycles .... 1 (+1 per indirect reference)
Reference pages 2-9, 3-1, 3-2, 3-35

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0067:a	Branch and Link — Unrestricted	J,P	BLU



The next sequential address (PROGRAM ADDRESS +1) replaces the contents of registerJ and the contents of register P (current PROGRAM ADDRESS) are replaced by the 5-bit immediate memory address.

NOTE

If the CPU is equipped with the Program Restrict system, the Program Restricted Flag (PRF) comes into consideration. Execution of the BLU instruction will turn OFF the PRF. If the computer is in a HALT condition and the PRF is ON, the BLU instruction will be treated as a NOP instruction.

Cycles 1
Reference pages 2-12, 3-1, 3-2, 3-35

3-8 COMPARE INSTRUCTIONS

The Compare group of instructions is composed of two basic types of operations: algebraic and logical comparisions. Both types of instructions compare two referenced quantities and set the Condition register according to the result. Algebraic comparisons treat the references as signed (+ or -) quantities, while logical comparisons assume the references are unsigned quantities.

Algebraic comparisons are identified by the letter "C" as the first letter in the instruction mnemonic (e.g., CAI). Logical comparisons use a mnemonic code beginning with the letter "K" (KAI). The second letter of the mnemonic code designates the first of the compared quantities (R_1) and the last letter designates the second quantity (R_2). For example:

C M I

Algebraically Compare (Type of operation) — C

Memory (R_1) — M

Register I (R_2) — I

or

K J A

Logically Compare (Type of operation) — K

Register J (R_1) — J

Register A (R_2) — A

Both algebraic and logical comparisons are performed according to the formula:

$$R_2 - R_1 = C \text{ (positive, zero or negative)}$$

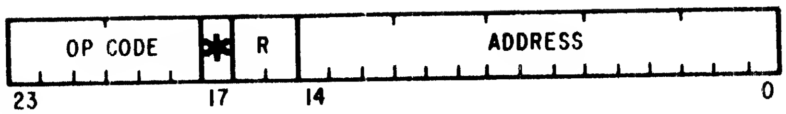
Therefore, $R_2 > R_1$, $R_2 < R_1$ and $R_2 = R_1$ will set the Condition register (C) to positive (+), negative (-) and zero (0), respectively.

The following instructions are included in the Series 6000 Compare group.

MNEMONIC	INSTRUCTION	PAGE
CAE	Compare A and E	3-50
CAI	Compare A and I	3-50
CAJ	Compare A and J	3-50
CAK	Compare A and K	3-50
CAT	Compare A and T	3-50
CEA	Compare E and A	3-50
CEI	Compare E and I	3-50
CEJ	Compare E and J	3-50

MNEMONIC	INSTRUCTION	PAGE
CEK	Compare E and K	3-50
CET	Compare E and T	3-50
CIA	Compare I and A	3-50
CIE	Compare I and E	3-50
CIJ	Compare I and J	3-50
CIK	Compare I and K	3-50
CIT	Compare I and T	3-50
CJA	Compare J and A	3-50
CJE	Compare J and E	3-50
CJI	Compare J and I	3-50
CJK	Compare J and K	3-50
CJT	Compare J and T	3-50
CKA	Compare K and A	3-50
CKE	Compare K and E	3-50
CKI	Compare K and I	3-50
CKJ	Compare K and J	3-50
CKT	Compare K and T	3-50
CMA	Compare Memory and A	3-46
CMB	Compare Memory and Byte	3-47
CME	Compare Memory and E	3-46
CMI	Compare Memory and I	3-46
CMJ	Compare Memory and J	3-46
CMK	Compare Memory and K	3-46
COB	Compare Operand and Byte	3-48
CTA	Compare T and A	3-50
CTE	Compare T and E	3-50
CTI	Compare T and I	3-50
CTJ	Compare T and J	3-50
CTK	Compare T and K	3-50
CZA	Compare Zero and A	3-48
CZD	Compare Zero and Double	3-49
CZE	Compare Zero and E	3-48
CZI	Compare Zero and I	3-48
CZJ	Compare Zero and J	3-48
CZK	Compare Zero and K	3-48
CZM	Compare Zero and Memory	3-47
CZT	Compare Zero and T	3-48
KAE	Kompare A and E	3-51
KAI	Kompare A and I	3-51
KAJ	Kompare A and J	3-51
KAK	Kompare A and K	3-51
KAT	Kompare A and T	3-51
KEA	Kompare E and A	3-51
KEI	Kompare E and I	3-51
KEJ	Kompare E and J	3-51
KEK	Kompare E and K	3-51
KET	Kompare E and T	3-51
KIA	Kompare I and A	3-51
KIE	Kompare I and E	3-51
KIJ	Kompare I and J	3-51

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
31. *+1:a	Compare Memory and I	C	CMI
31. *+2:a	J	C	CMJ
31. *+3:a	K	C	CMK



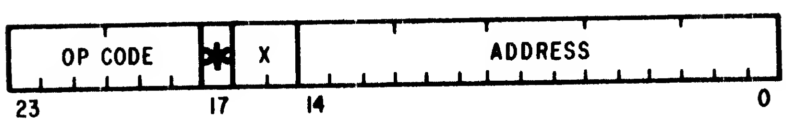
The contents of the effective memory address and the contents of register I, J or K are algebraically compared and the Condition register is set to the status of the result.

NOTE

The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-43

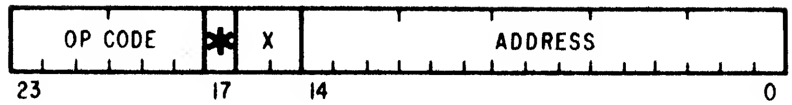
32. *+X:a	Compare Memory and A	C	CME
33. *+X:a	E	C	CMA



The contents of the effective memory address and the contents of register E or A are algebraically compared and the Condition register is set to the status of the result.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-43

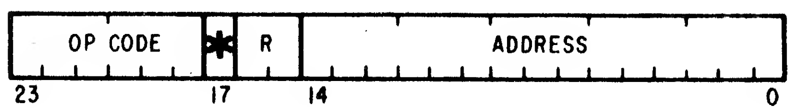
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
34. *+X:a	Compare Memory and Byte	C	CMB



The contents of register B (A_0-A_7) and the contents of the effective memory address (M_0-M_7) are algebraically compared and the Condition register is set to the status of the result.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-43

41. *+0:a	Compare Zero and Memory	C	CZM



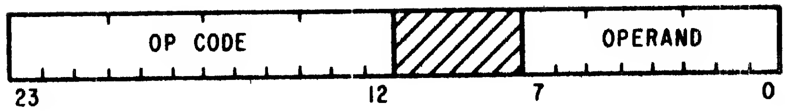
The contents of the effective memory address and zero are algebraically compared and the Condition register is set to the status of the result.

NOTE

The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-43

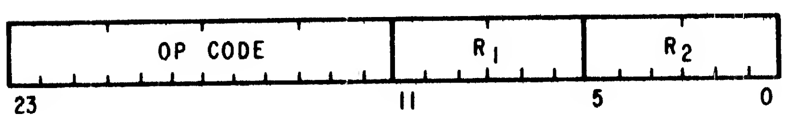
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0014:o	Compare Operand and Byte	C	COB



The 8-bit signed operand and the contents of register B (A_0-A_7) are algebraically compared and the Condition register is set to the status of the result.

Cycles 1
Reference pages 3-1, 3-43

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0024.00.01	Compare Zero and I	C	CZI
0024.00.02	J	C	CZJ
0024.00.04	K	C	CZK
0024.00.10	E	C	CZE
0024.00.20	A	C	CZA
0024.00.40	T	C	CZT



The contents of the specified register and zero are algebraically compared and the Condition register is set to the status of the result.

Cycles 1
Reference pages 3-1, 3-43

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0024.00.30	Compare Zero and Double	C	CZD

The contents of register D (E and A) and zero are algebraically compared and the Condition register is set to the status of the result.

Cycles 1
Reference pages 3-1, 3-43

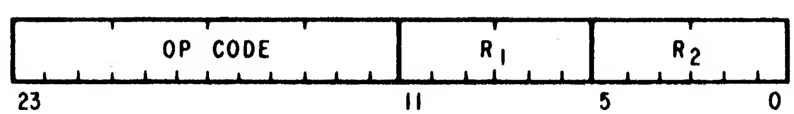
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0024.01.02	Compare I and J	C	CIJ
0024.01.04	K	C	CIK
0024.01.10	E	C	CIE
0024.01.20	A	C	CIA
0024.01.40	T	C	CIT
0024.02.01	Compare J and I	C	CJI
0024.02.04	K	C	CJK
0024.02.10	E	C	CJE
0024.02.20	A	C	CJA
0024.02.40	T	C	CJT
0024.04.01	Compare K and I	C	CKI
0024.04.02	and J	C	CKJ
0024.04.10	and E	C	CKE
0024.04.20	and A	C	CKA
0024.04.40	and T	C	CKT
0024.10.01	Compare E and I	C	CEI
0024.10.02	J	C	CEJ
0024.10.04	K	C	CEK
0024.10.20	A	C	CEA
0024.10.40	T	C	CET
0024.20.01	Compare A and I	C	CAI
0024.20.02	J	C	CAJ
0024.20.04	K	C	CAK
0024.20.10	E	C	CAE
0024.20.40	T	C	CAT
0024.40.01	Compare T and I	C	CTI
0024.40.02	J	C	CTJ
0024.40.04	K	C	CTK
0024.40.10	E	C	CTE
0024.40.20	A	C	CTA

OP CODE	R_1	R_2
23	11	5 0

The contents of R_1 and the contents of R_2 are algebraically compared and the Condition register is set to the status of the result.

Cycles 1
Reference pages 3-1, 3-43

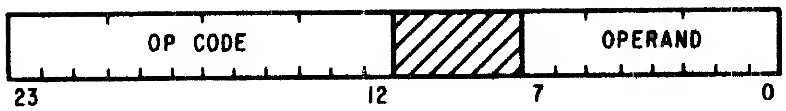
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0025.01.02	Kompare I and J	C	KIJ
0025.01.04	and K	C	KIK
0025.01.10	and E	C	KIE
0025.01.20	and A	C	KIA
0025.01.40	and T	C	KIT
0025.02.01	Kompare J and I	C	KJI
0025.02.04	and K	C	KJK
0025.02.10	and E	C	KJE
0025.02.20	and A	C	KJA
0025.02.40	and T	C	KJT
0025.04.01	Kompare K and I	C	KKI
0025.04.02	and J	C	KKJ
0025.04.10	and E	C	KKE
0025.04.20	and A	C	KKA
0025.04.40	and T	C	KKT
0025.10.01	Kompare E and I	C	KEI
0025.10.02	and J	C	KEJ
0025.10.04	and K	C	KEK
0025.10.20	and A	C	KEA
0025.10.40	and T	C	KET
0025.20.01	Kompare A and I	C	KAI
0025.20.02	and J	C	KAJ
0025.20.04	and K	C	KAK
0025.20.10	and E	C	KAE
0025.20.40	and T	C	KAT
0025.40.01	Kompare T and I	C	KTI
0025.40.02	and J	C	KTJ
0025.40.04	and K	C	KTK
0025.40.10	and E	C	KTE
0025.40.20	and A	C	KTA



The contents of R_1 and R_2 are logically compared and the Condition register is set according to the result.

Cycles 1
Reference pages 3-1, 4-43

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0015:o	Kompare Operand and Byte	C	KOB



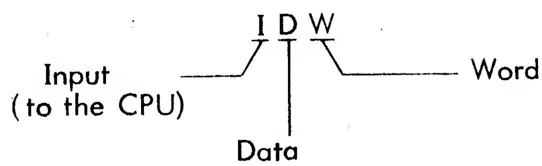
The 8-bit operand and the contents of register B (A_0-A_7) are logically compared and the Condition register is set according to the result.

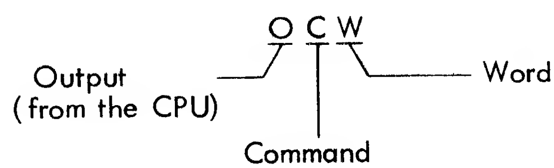
Cycles 1
Reference pages 3-1, 3-43

3-9 INPUT/OUTPUT INSTRUCTIONS

The Input/Output (I/O) instructions provide the required control for all communications between the Series 6000 CPU and the input/output structure. In addition to controlling data transfers between the CPU and peripheral units, the I/O instruction repertoire allows peripheral unit command functions and status testing to be placed under program control.

The specific I/O operation can be identified by examination of the individual instruction mnemonics. All I/O instruction mnemonics use the letter "W" to indicate that a full word is to be transferred between the CPU and the I/O syructure. The first letter of the mnemonic indicates the direction of the transfer (input or output). The second letter indicates the type of word to be transferred. For example:





There is no "I/O hold", or delay, imposed by the hardware. All I/O instructions are executed unconditionally; i.e., the CPU is not forced to wait for a response from the I/O structure in order to complete the instruction execution cycle.

Although there is no built-in hold/delay provision, a programmed delay can be implemented if desired. At the beginning of each I/O instruction cycle, the Condition register is cleared. At the end of the execution phase of each I/O instruction, bit 2 (ZERO) is set to "ZERO" if the selected channel was ready and accepted the command. If the selected channel was not ready, bit 2 of the Condition register remains set to "NOT ZERO". The program can test the "NOT ZERO" state of bit 2 with a branch instruction following the I/O instruction. When bit 2 is set to "NOT ZERO+, a programmed delay is implemented. For example:

```
ODW    '0103     Output word to Channel 1, Unit 3
BNZ    *-1       Delay if not ready
---              Continue if ready
```

Two examples of a channel being not ready are that the peripheral units data transfer capability is slower than that of the program loop and therefore cannot accept data as it is available from the channel. The other example occurs in a channel/multiunit environment where the channel is connected to peripheral unit A and peripheral unit B is selected for a data transfer. In this instance, the channel remains not ready until a disconnect/connect sequence is performed and peripheral unit B is connected to the channel. Two cycles are required for the disconnect/connect sequence.

NOTE

Status returned to the Condition register immediately after completion of an I/O instruction refers to channel status only. A ready ("ZERO") condition indicates the channel accepted the I/O command. This does not imply the I/O operation was completed with the selected peripheral unit.

If the program selects a non-existant channel or unit, the channel accepts the command or data and leaves bit 2 of the Condition register set to "NOT ZERO" to indicate not ready. The channel will remain not ready for any subsequent commands.

If the Series 6000 system is equipped with the Program Restrict/Instruction Trap option, all I/O instructions will be affected.

The following instructions are included in the Input/Output group.

MNEMONIC	INSTRUCTION	PAGE
IAW	Input Address Word	3-61
IDW	Input Data Word	3-58
ISW	Input Status Word	3-56
OAW	Output Address Word	3-60
OCW	Output Command Word	3-55
ODW	Output Data Word	3-57

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0070. *+C. U	Output Command Word	C	OCW

OP CODE	*	CHANNEL		UNIT
23	11		5	3 0

A 24-bit command word is transferred from the A register to the specified channel/unit combination and the Condition register is set to "ZERO".

If the selected channel is not ready, the Condition register remains set to "NOT ZERO" which allows a programmed delay if desired.

If the Override bit (*) is set (ONE), the command word assumes immediate control over the channel. The contents of the A register are transferred to the channel and a disconnect/connect sequence is initiated. The Condition register is set to "ZERO" to indicate the channel has accepted but not necessarily executed the command. Upon completion of the disconnect/connect sequence, the channel transfers the command word to the unit.

If the Override bit is not set (ZERO) and the OCW specifies a unit other than the unit connected to the channel and the channel is ready, the command word is accepted by the channel. The Condition register is set to "NOT ZERO" to indicate the channel is not ready. A disconnect/connect sequence is performed and the command is transferred to the unit. The Condition register is reset to "ZERO" to indicate ready.

NOTE

> Following the execution of an OCW, the channel remains not ready until the peripheral unit accepts the data.

If the selected channel is an ABC channel and is actively engaged in a block transfer, executing an OCW with the Override bit set terminates the transfer sequence leaving the contents of the TAR and WCR intact. If the Override bit is not set and the ABC channel is engaged in a block transfer, the OCW instruction will be ignored. The Condition register will remain set to "NOT ZERO". Once an ABC channel is activated it will not accept an OCW with the Override bit not set until the word count is complete; i.e., all words in the block have been transferred and WCR equals zero.

Cycles 1
Reference pages 3-1, 3-53

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0073. 00+C. U	Input Status Word	A,C	ISW

OP CODE	0	CHANNEL		UNIT
23	11	5	3	0

A status word is transferred from the specified channel/unit combination to register A and the Condition register is set to "Zero".

If the addressed channel/unit combination is not ready (see notes 1 and 2) or a status word is not available, the Condition register is set to "Not Zero" to allow a programmed delay.

NOTES

(1) If the selected channel is in the process of executing a command (resulting from a previous OCW), the channel indicates not ready (Condition register remains set to "NOT ZERO") and ignores the ISW instruction until the peripheral unit accepts the OCW command. The channel indicates ready (Condition register set to "ZERO") and accepts the ISW when it is executed again.

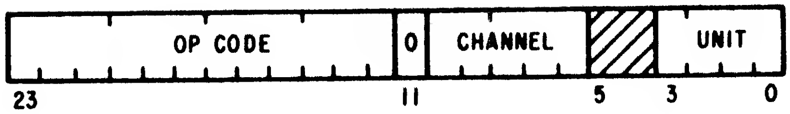
(2) If the ISW specifies a unit other than the unit connected to the channel, the channel indicates not ready and ignores the command. A disconnect/connect is initiated.

(3) If the selected channel is an ABC channel engaged in a block transfer, the Condition register is set to "ZERO" and a 24-bit status word is transferred to the A register. Bits 0 through 7 contain the unit status and bit 23 contains the ABC word count status.

(4) If the selected unit is receiving data as the result of an ODW instruction, the ISW will be accepted and the Condition register is set to "ZERO".

Cycles 1
Reference pages 3-1, 3-53

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0071.00+C.U	Output Data Word	C	ODW



A data word is transferred from register A to the specified channel/unit combination and the Condition register is set to "Zero".

If the channel is busy and cannot accept the data word, the Condition register is set to "Not Zero" to allow a programmed delay.

NOTES

(1) Although a 24-bit word is transferred to the channel, the peripheral unit accepts only a predetermined number of bits (dictated by peripheral unit design).

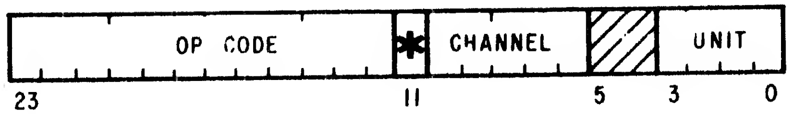
(2) For character-oriented units and units accepting data words of less than 24 bits, the data for transfer must be right-justified in the A register prior to executing the ODW instruction.

(3) If ODW instruction specifies a unit other than the unit connected to the channel and the channel is ready, the channel accepts the ODW, sets the Condition register to "ZERO", and initiates a disconnect/connect sequence. After completion of the disconnect/connect sequence, the ODW is transferred to the unit. The channel indicates ready to subsequent I/O instructions.

(4) If the ODW instruction specifies an ABC channel that is engaged in a block transfer, the Condition register remains set to "NOT ZERO" and the ODW is ignored. An ABC channel, once activated, will not accept an ODW instruction until the word count is complete; i.e., all words in the block have been transferred and WCR equals zero.

Cycles 1
Reference pages 3-1, 3-53

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0072. *+C.U	Input Data Word	A,C	IDW



A data word is transferred from the specified channel/unit combination to register A and the Condition register is set to "Zero".

If the channel is not ready or data from the specified unit is not available, the Condition register is set to "Not Zero" to allow a programmed delay.

NOTES

(1) If the selected unit is in the process of executing a command as the result of a previous OCW instruction, the channel indicates not ready (Condition register remains set to "NOT ZERO") and the IDW is ignored. At the completion of the OCW, the Condition register is set to "ZERO" and the IDW instruction will be accepted by the channel.

(2) If the selected unit is in the process of receiving data as a result of an ODW instruction and data is available from the unit, an ODW will be accepted and the Condition register set to "ZERO".

(3) If the IDW instruction specifies a unit other than the unit connected to the channel, the channel indicates not ready (Condition register remains set to "NOT ZERO"), ignores the instruction, and initiates a disconnect/connect sequence.

(4) If a IDW instruction specifies an ABC channel that is engaged in a block transfer, the Condition register remains set to "NOT ZERO" (channel not ready) and the instruction is ignored. An ABC channel, once activated, will not accept an IDW instruction until the word count is complete; i.e., all words in the block have been transferred and WCR equals zero.

If the Merge bit (*) is ZERO, register A is cleared prior to the data transfer. Input data is right-justified in register A.

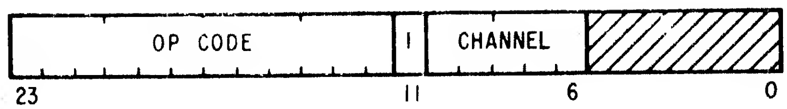
If the Merge bit is a ONE, an OR is performed between the previous contents of register A and the incoming data word. This feature, in conjunction with a shift operation, allows input data characters to be packed in register A.

Example: Two 12-bit data characters are to be packed in register A.

IDW	'0102	Clear A and load first character from Channel 01, Unit 02
BNZ	*-1	Wait if busy
LLA	12	Shift the contents of A left 12 bits
IDW*	'0102	Merge second character
BNZ	*-1	Wait if busy
...		Continue

Cycles 1
Reference pages 3-1, 3-53

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0071.40+C	Output Address Word	C	OAW



The contents of register A are transferred to the Transfer Address Register (TAR) in the specified ABC channel. The Condition register is set to "Zero".

NOTE

> An ABC channel will always indicate ready for an OAW instruction. However, if the OAW specifies an invalid channel number, it will receive a "not ready" indication and the Condition register remains set to "Not Zero".

The OAW instruction does not activate an ABC channel. It transfers the starting memory address of the first of two ABC parameter words from the A register to the TAR in the selected ABC channel.

If an OAW instruction addresses an ABC channel during a block transfer sequence, the sequence will be terminated.

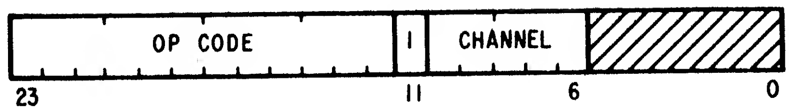
If the OAW instruction addresses a standard channel, the Condition register remains set to "NOT ZERO" indicating channel not ready.

NOTE

> Refer to paragraph 4-4.5 for a description of an Automatic Block Transfer.

Cycles 1
Reference pages 3-1, 3-53

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0073.40+C	Input Address Word	A,C	IAW



The current contents of the Transfer Address Register (TAR) in the specified ABC channel are transferred to register A and the Condition register is set to "Zero".

NOTES

(1) An ABC channel always indicates ready to an IAW instruction.

(2) If the IAW instruction specifies an invalid channel or a standard channel, the Condition register remains set to "NOT ZERO" indicating channel not ready.

(3) Refer to paragraph 4-4.5 for a description of an Automatic Block Transfer.

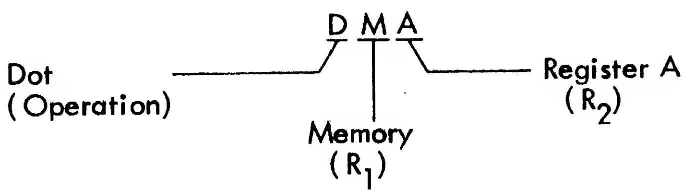
Cycles 1
Reference pages 3-1, 3-53

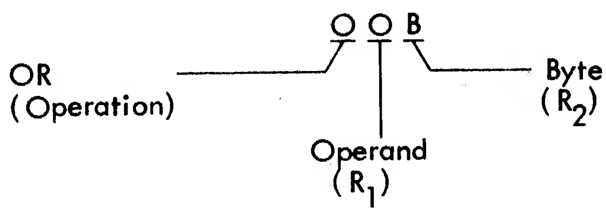
This Page Left Blank Intentionally

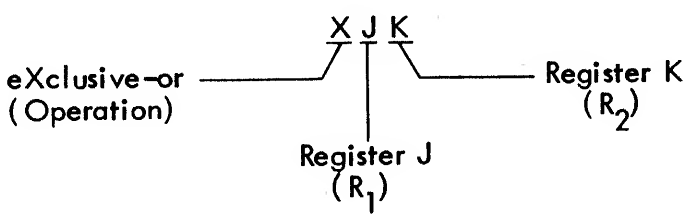
3-10 LOGICAL INSTRUCTIONS

The Logical group of instructions includes AND (DOT product), OR and EXCLUSIVE-OR operations. All three types use two quantities to produce a logical result. The AND instructions use a mnemonic code beginning with the letter "D", for "Dot". The OR instructions use a mnemonic beginning with the letter "O", while EXCLUSIVE-OR instructions are distinguished by the letter "X".

The second letter of the mnemonic code identifies the first of the two quantities (R_1). The third letter signifies the second quantity (R_2). Some examples are listed below:







Unless specifically noted otherwise in the individual descriptions, the result of the logical operation replaces the previous contents of R_2 while R_1 is unchanged. The Condition register is set to the status of the result (Positive, Negative or Zero) after the operation. The various logical operations are illustrated in the following table.

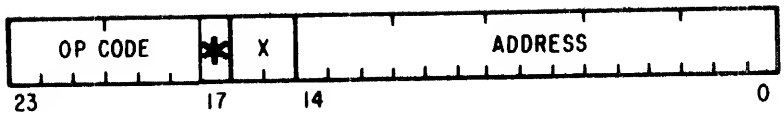
R_1	R_2	R_1 .AND. R_2	R_1 .OR. R_2	R_1 .XOR. R_2
1	1	1	1	0
0	1	0	1	1
1	0	0	1	1
0	0	0	0	0

The following instructions are included in the Logical group.

MNEMONIC	INSTRUCTION	PAGE
DAE	Dot A with E	3-67
DAI	Dot A with I	3-67
DAJ	Dot A with J	3-67
DAK	Dot A with K	3-67
DAT	Dot A with T	3-67
DEA	Dot E with A	3-67
DEI	Dot E with I	3-67
DEJ	Dot E with J	3-67
DEK	Dot E with K	3-67
DET	Dot E with T	3-67
DIA	Dot I with A	3-67
DIE	Dot I with E	3-67
DIJ	Dot I with J	3-67
DIK	Dot I with K	3-67
DIT	Dot I with T	3-67
DJA	Dot J with A	3-67
DJE	Dot J with E	3-67
DJI	Dot J with I	3-67
DJK	Dot J with K	3-67
DJT	Dot J with T	3-67
DKA	Dot K with A	3-67
DKE	Dot K with E	3-67
DKI	Dot K with I	3-67
DKJ	Dot K with J	3-67
DKT	Dot K with T	3-67
DMA	Dot Memory with A	3-66
DOB	Dot Operand with Byte	3-66
DTA	Dot T with A	3-67
DTE	Dot T with E	3-67
DTI	Dot T with I	3-67
DTJ	Dot T with J	3-67
DTK	Dot T with K	3-67
OAE	Or A with E	3-69
OAI	Or A with I	3-69
OAJ	Or A with J	3-69
OAK	Or A with K	3-69
OAT	Or A with T	3-69
OEA	Or E with A	3-69
OEI	Or E with I	3-69
OEJ	Or E with J	3-69
OEK	Or E with K	3-69
OET	Or E with T	3-69
OIA	Or I with A	3-69
OIE	Or I with E	3-69
OIJ	Or I with J	3-69
OIK	Or I with K	3-69
OIT	Or I with T	3-69
OJA	Or J with A	3-69
OJE	Or J with E	3-69

MNEMONIC	INSTRUCTION	PAGE
OJI	Or J with A	3-69
OJK	Or J with K	3-69
OJT	Or J with T	3-69
OKA	Or K with A	3-69
OKE	Or K with E	3-69
OKI	Or K with I	3-69
OKJ	Or K with J	3-69
OKT	Or K with T	3-69
OMA	Or Memory with A	3-68
OOB	Or Operand with Byte	3-68
OTA	Or T with A	3-69
OTE	Or T with E	3-69
OTI	Or T with I	3-69
OTJ	Or T with J	3-69
OTK	Or T with K	3-69
XAE	eXclusive-or A with E	3-71
XAI	eXclusive-or A with I	3-71
XAJ	eXclusive-or A with J	3-71
XAK	eXclusive-or A with K	3-71
XAT	eXclusive-or A with T	3-71
XEA	eXclusive-or E with A	3-71
XEI	eXclusive-or E with I	3-71
XEJ	eXclusive-or E with J	3-71
XEK	eXclusive-or E with K	3-71
XET	eXclusive-or E with T	3-71
XIA	eXclusive-or I with A	3-71
XIE	eXclusive-or I with E	3-71
XIJ	eXclusive-or I with J	3-71
XIK	eXclusive-or I with K	3-71
XIT	eXclusive-or I with T	3-71
XJA	eXclusive-or J with A	3-71
XJE	eXclusive-or J with E	3-71
XJI	eXclusive-or J with I	3-71
XJK	eXclusive-or J with K	3-71
XJT	eXclusive-or J with T	3-71
XKA	eXclusive-or K with A	3-71
XKE	eXclusive-or K with E	3-71
XKI	eXclusive-or K with I	3-71
XKJ	eXclusive-or K with J	3-71
XKT	eXclusive-or K with T	3-71
XMA	eXclusive-or Memory with A	3-70
XOB	eXclusive-or Operand with Byte	3-70
XTA	eXclusive-or T with A	3-71
XTE	eXclusive-or T with E	3-71
XTI	eXclusive-or T with I	3-71
XTJ	eXclusive-or T with J	3-71
XTK	eXclusive-or T with K	3-71

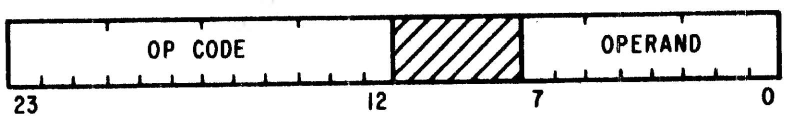
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
36. *+X:a	Dot Memory with A	A,C	DMA



A logical AND is performed between the contents of the effective memory address and the contents of register A.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-63

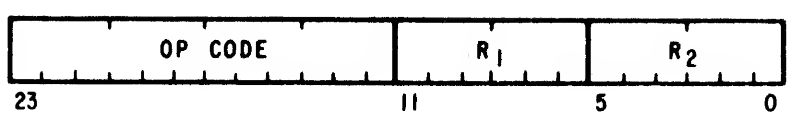
0016:o	Dot Operand with Byte	A,C	DOB



A logical AND is performed between the 8-bit operand and the contents of register B (A_0-A_7). Bits A_8-A_{23} are unchanged.

Cycles 1
Reference pages 3-1, 3-63

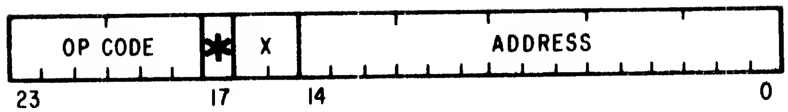
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0026.01.02	Dot I with J	J,C	DIJ
0026.01.04	K	K,C	DIK
0026.01.10	E	E,C	DIE
0026.01.20	A	A,C	DIA
0026.01.40	T	T,C	DIT
0026.02.01	Dot J with I	I,C	DJI
0026.02.04	K	K,C	DJK
0026.02.10	E	E,C	DJE
0026.02.20	A	A,C	DJA
0026.02.40	T	T,C	DJT
0026.04.01	Dot K with I	I,C	DKI
0026.04.02	J	J,C	DKJ
0026.04.10	E	E,C	DKE
0026.04.20	A	A,C	DKA
0026.04.40	T	T,C	DKT
0026.10.01	Dot E with I	I,C	DEI
0026.10.02	J	J,C	DEJ
0026.10.04	K	K,C	DEK
0026.10.20	A	A,C	DEA
0026.10.40	T	T,C	DET
0026.20.01	Dot A with I	I,C	DAI
0026.20.02	J	J,C	DAJ
0026.20.04	K	K,C	DAK
0026.20.10	E	E,C	DAE
0026.20.40	T	T,C	DAT
0026.40.01	Dot T with I	I,C	DTI
0026.40.02	J	J,C	DTJ
0026.40.04	K	K,C	DTK
0026.40.10	E	E,C	DTE
0026.40.20	A	A,C	DTA



A logical AND is performed between the contents of R_1 and R_2.

Cycles 1
Reference pages 3-1, 3-63

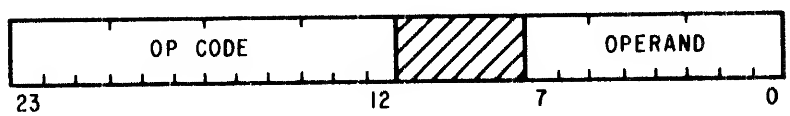
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
35. *+X:a	Or Memory with A	A,C	OMA



A logical OR is performed between the contents of the effective memory address and the contents of register A.

Cycles . . . . 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-63

0004:o	Or Operand with Byte	A,C	OOB



A logical OR is performed between the 8-bit operand and the contents of register B (A_0-A_7). Bits A_8-A_{23} are unchanged.

Cycles 1
Reference pages 3-1, 3-63

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0030.03.02	Or I with J	J,C	OIJ
0030.05.04	K	K,C	OIK
0030.11.10	E	E,C	OIE
0030.21.20	A	A,C	OIA
0030.41.40	T	T,C	OIT
0030.03.01	Or J with I	I,C	OJI
0030.06.04	K	K,C	OJK
0030.12.10	E	E,C	OJE
0030.22.20	A	A,C	OJA
0030.42.40	T	T,C	OJT
0030.05.01	Or K with I	I,C	OKI
0030.06.02	J	J,C	OKJ
0030.14.10	E	E,C	OKE
0030.24.20	A	A,C	OKA
0030.44.40	T	T,C	OKT
0030.11.01	Or E with I	I,C	OEI
0030.12.02	J	J,C	OEJ
0030.14.04	K	K,C	OEK
0030.30.20	A	A,C	OEA
0030.50.40	T	T,C	OET
0030.21.01	Or A with I	I,C	OAI
0030.22.02	J	J,C	OAJ
0030.24.04	K	K,C	OAK
0030.30.10	E	E,C	OAE
0030.60.40	T	T,C	OAT
0030.41.01	Or T with I	I,C	OTI
0030.42.02	J	J,C	OTJ
0030.44.04	K	K,C	OTK
0030.50.10	E	E,C	OTE
0030.60.20	A	A,C	OTA

OP CODE	$R_1 + R_2$	R_2
23	11	5 0

A logical OR is performed between the contents of R_1 and R_2.

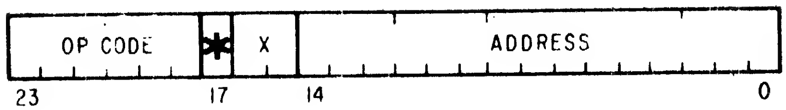
NOTE

The general instruction formula for this group of Logical instructions is:

$$0030.\ R_1 + R_2 \cdot R_2$$

Cycles **1**
Reference pages **3-1, 3-63**

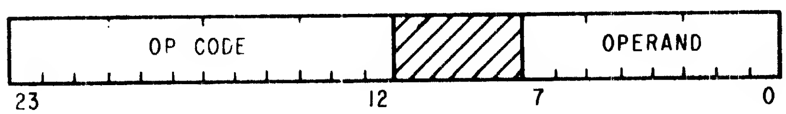
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
37. *+X:a	eXclusive-or Memory with A	A,C	XMA



An EXCLUSIVE-OR operation is performed between the contents of the effective memory address and the contents of register A.

Cycles .... 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-63

0017:o	eXclusive-or Operand with Byte	A,C	XOB



An EXCLUSIVE-OR operation is performed between the 8-bit operand and the contents of register B (A_0-A_7). Bits A_8-A_{23} are unchanged.

Cycles 1
Reference pages 3-1, 3-63

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0027.01.02	eXclusive-or I with J	J,C	XIJ
0027.01.04	K	K,C	XIK
0027.01.10	E	E,C	XIE
0027.01.20	A	A,C	XIA
0027.01.40	T	T,C	XIT
0027.02.01	eXclusive-or J with I	I,C	XJI
0027.02.04	K	K,C	XJK
0027.02.10	E	E,C	XJE
0027.02.20	A	A,C	XJA
0027.02.40	T	T,C	XJT
0027.04.01	eXclusive-or K with I	I,C	XKI
0027.04.02	J	J,C	XKJ
0027.04.10	E	E,C	XKE
0027.04.20	A	A,C	XKA
0027.04.40	T	T,C	XKT
0027.10.01	eXclusive-or E with I	I,C	XEI
0027.10.02	J	J,C	XEJ
0027.10.04	K	K,C	XEK
0027.10.20	A	A,C	XEA
0027.10.40	T	T,C	XET
0027.20.01	eXclusive-or A with I	I,C	XAI
0027.20.02	J	J,C	XAJ
0027.20.04	K	K,C	XAK
0027.20.10	E	E,C	XAE
0027.20.40	T	T,C	XAT
0027.40.01	eXclusive-or T with I	I,C	XTI
0027.40.02	J	J,C	XTJ
0027.40.04	K	K,C	XTK
0027.40.10	E	E,C	XTE
0027.40.20	A	A,C	XTA

OP CODE	R_1	R_2
23 – 12	11 – 6	5 – 0

An EXCLUSIVE-OR function is performed between the contents of R_1 and R_2.

Cycles 1
Reference pages 3-1, 3-63

This Page Left Blank Intentionally.

3-11 PRIORITY INTERRUPT INSTRUCTIONS

The Priority Interrupt instruction group provides the means for program control of external interrupts. External interrupts may be selectively armed, disarmed, enabled or inhibited under program control. Other instructions provide the means for holding and releasing external interrupts, while others are available for transferring control upon interrupt detection. For a detailed description of the Series 6000 Priority Interrupt System, refer to Section II of this manual.

If the Series 6000 system is equipped with the optional Program Restrict System and Instruction Trap, the following Priority Interrupt instructions will be affected:

a) Hold eXternal Interrupts (HXI)
b) Release eXternal Interrupts (RXI)
c) Unitarily Arm group 1 interrupts (UA1)
d) Unitarily Arm group 2 interrupts (UA2)
e) Unitarily Arm group 3 interrupts (UA3)
f) Unitarily Disarm group 1 interrupts (UD1)
g) Unitarily Disarm group 2 interrupts (UD2)
h) Unitarily Disarm group 3 interrupts (UD3)
i) Unitarily Enable group 1 interrupts (UE1)
j) Unitarily Enable group 2 interrupts (UE2)
k) Unitarily Enable group 3 interrupts (UE3)
l) Unitarily Inhibit group 1 interrupts (UI1)
m) Unitarily Inhibit group 2 interrupts (UI2)
n) Unitarily Inhibit group 3 interrupts (UI3)
o) Transfer Double to group 1 (TD1)
p) Transfer Double to group 2 (TD2)
q) Transfer Double to group 3 (TD3)

The following instructions are included in the Priority Interrupt group:

MNEMONIC	INSTRUCTION	PAGE
BRL	Branch and Reset interrupt — Long	3-76
BSL	Branch and Save Return — Long	3-75
HTI	Hold interrupts & Transfer I to memory	3-77
HTJ	Hold interrupts & Transfer J to memory	3-77
HTK	Hold interrupts & Transfer K to memory	3-77
HXI	Hold eXternal Interrupts	3-77
RXI	Release eXternal Interrupts	3-78
T1D	Transfer group 1 to Double	3-79
T2D	Transfer group 2 to Double	3-79
T3D	Transfer group 3 to Double	3-79
TD1	Transfer Double to group 1	3-78
TD2	Transfer Double to group 2	3-78
TD3	Transfer Double to group 3	3-78
UA1	Unitarily Arm group 1	3-80
UA2	Unitarily Arm group 2	3-80
UA3	Unitarily Arm group 3	3-80
UD1	Unitarily Disarm group 1	3-81
UD2	Unitarily Disarm group 2	3-81
UD3	Unitarily Disarm group 3	3-81

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
25. *+0:A	Branch and Save return — Long	P	BSL

OP CODE	*	0	ADDRESS
23		15	0

The next sequential address (PROGRAM ADDRESS +1), along with the contents of the Condition register, are stored in the 16-bit effective memory address (EMA). The contents of register P (current PROGRAM ADDRESS) are then replaced by the address following the effective memory address (EMA +1).

This instruction is normally used to enter an interrupt subroutine because it provides a means of returning to the main program at the point of interrupt and saves the machine status (Condition) at the time of the interrupt.

NOTES

(1) The contents of the Condition register are stored in bit positions 16-19 of the EMA and the return address (PROGRAM ADDRESS H) is stored in bits 0-15. The remaining bits are set to ZEROs, however, refer to note 2 for variation on bit 20.

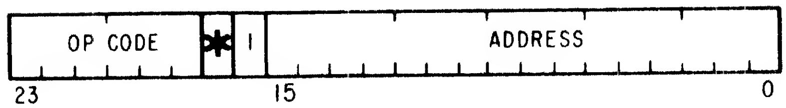
(2) If a "Power Down" interrupt occurs when the CPU is halted, bit 20 is set to ZERO. If the CPU is running when the "Power Down" interrupt occurs, bit 20 is set to a ONE.

(3) The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

(4) External interrupts are prohibited for the period of one instruction following the execution of this instruction.

Cycles .... 2 (+1 per indirect reference)
Reference pages 2-9, 3-1, 3-2, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
25. *+2:A	3ranch and Reset Interrupt — Long	C, P	BRL



The highest-level active interrupt is reset (i.e., returned to the inactive state) and the contents of register P (current PROGRAM ADDRESS) are replaced by the 16-bit effective memory address.

BRL is normally used to exit an interrupt subroutine. If BRL contains an indirect reference, the last word in the indirect address chain contains the previous status (i.e., C register contents at the time of the interrupt) in bit positions M_{16}-M_{19} and the return address in bit positions M_0-M_{15} as a result of the BSL instruction. The C register is restored and the program branches to the return address (restarting the machine to the preinterrupt status).

Example:

```
        .
        .
        TMA
L       AMA
        SMA            Interrupt occurs (EXM K).
        .
K       BSL   M        Dedicated interrupt location.

M       ***            M becomes L+1 as a result of BSL at K.
        .              The C register contents are stored in
        .              M16-M19.
        BRL*           Restore C register and return to L+1.
```

NOTES

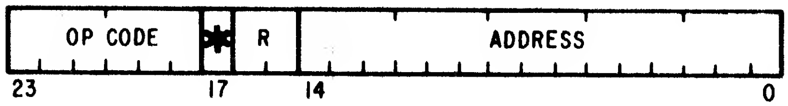
(1) The BRL will not reset the interrupt if external interrupts have been held by an HXI instruction. Control will be returned to the effective memory address.

(2) Executive traps, which are not affected by the HXI instruction, will be reset by the BRL.

(3) The immediate memory reference cannot be indexed; however, indexing indirect references is permitted.

Cycles .... 1 (+1 per indirect reference)
Reference pages 2-9, 3-1, 3-2, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
27. *+1:a	Hold interrupts and Transfer I to Memory	M	HTI
27. *+2:a	Transfer J to Memory	M	HTJ
27. *+3:a	Transfer K to Memory	M	HTK



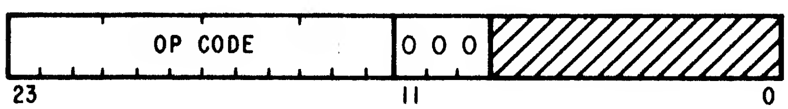
The contents of register I, J or K replace the previous contents of the effective memory address and external interrupts are prohibited for the period of one instruction following the execution of this instruction.

NOTE

> The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-71

0066. 0	Hold eXternal Interrupts		HXI



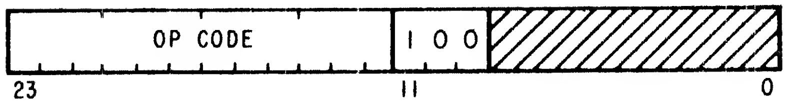
The activation of any external interrupt is prohibited. The prohibition is effective immediately upon execution of the instruction and lasts until the interrupts are released (see RXI instruction). Both the address trap and interval timer traps are prohibited from becoming active while the HXI is in effect.

NOTE

> Only the two executive traps mentioned are affected by this instruction.

Cycles 1
Reference pages 2-9, 3-1, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0066.4	Release eXternal Interrupts		RXI

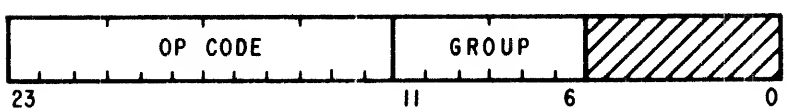


The prohibition imposed by the HXI instruction is removed, allowing any external interrupt to be activated 1 cycle after this instruction. This permits the next sequential instruction to be executed without external interruption.

If an address trap or timer trap had been triggered while an HXI was in effect, the highest level will come in first after the RXI instruction.

Cycles 1
Reference pages 2-9, 3-1, 3-71

0064.01	Transfer Double to group 1	1 A/D, 1 E/I	TD1
0064.02	group 2	2 A/D, 2 E/I	TD2
0064.04	group 3	3 A/D, 3 E/I	TD3



The contents of register D (E and A) replace the previous contents of the Arm/Disarm (A/D) and Enable/Inhibit (E/I) registers of the specified interrupt group (1, 2 or 3). The contents of E are transferred to the A/D register and the contents of A are transferred to the E/I register.

NOTE

The external interrupt structure is cleared by the execution of these instructions.

Cycles 1
Reference pages 2-9, 3-1, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0065.01	Transfer group 1 to Double	E,A	T1D
0065.02	group 2	E,A	T2D
0065.04	group 3	E,A	T3D

The contents of the Arm/Disarm (A/D) and Enable/Inhibit (E/I) registers of the specified interrupt group replace the previous contents of register D (E and A). The contents of the A/D register are transferred to register E and the contents of the E/I register are transferred to register A.

NOTE

The states of the external interrupts are not affected by the execution of these instructions.

Cycles 1
Reference pages 3-1, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0060.01	Unitarily Arm group 1 interrupts	1 A/D	UA1
0060.02	group 2	2 A/D	UA2
0060.04	group 3	3 A/D	UA3

Any number of the 24 interrupt levels in the specified group are selectively armed; i.e., the selected bit(s) of the Arm/Disarm (A/D) register are set to ONE.

The corresponding bit(s) of register A must be set to select the appropriate level(s) prior to executing these instructions.

Example: Arm levels 1 and 3, group 1

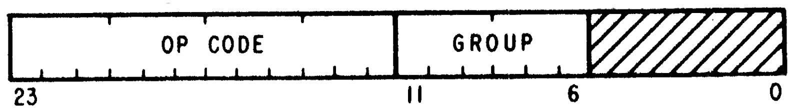
TOA	B1B3	Select levels 1 and 3 (set bits 1 and 3 of A)
UA1		Arm selected levels of group 1

NOTE

Execution of these instructions does not clear the interrupt structure and, therefore, does not affect any interrupt levels other than those selected. If a level selected for arming is already armed, it is not cleared by the execution of these instructions.

Cycles 1
Reference pages 2-9, 3-1, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0061.01	Unitarily Disarm group 1 interrupts	1 A/D	UD1
0061.02	group 2	2 A/D	UD2
0061.04	group 3	3 A/D	UD3



Any number of the 24 interrupt levels in the specified group are selectively disarmed; i.e., the selected bits of the Arm/Disarm (A/D) register are reset to ZERO.

The corresponding bit(s) of register A must be set to select the appropriate level(s) prior to executing these instructions.

Example: Disarm level 2, group 3

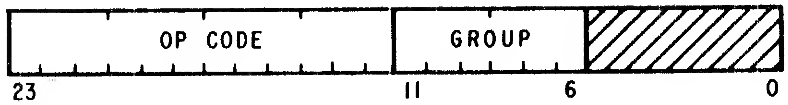
```
TOA   B2      Select level 2 (set bit 2 of A)
UD3           Disarm selected level of group 3
```

NOTE

Execution of these instructions will clear only those levels which are selected. The remaining levels will not be affected.

Cycles 1
Reference pages 2-9, 3-1, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0062.01	Unitarily Enable group 1 interrupts	1 E/I	UE1
0062.02	group 2	2 E/I	UE2
0062.04	group 3	3 E/I	UE3



Any number of the 24 interrupt levels in the specified group are selectively enabled; i.e., the selected bits of the Enable/Inhibit (E/I) register are set to ONE.

The corresponding bit(s) of register A must be set to select the appropriate level(s) prior to executing these instructions.

Example: Enable levels 0, 2 and 5, group 2

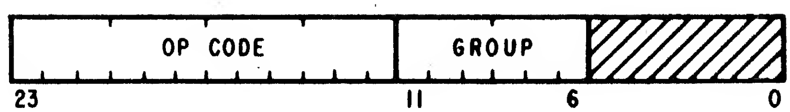
TOA	B0B2B5	Select levels 0, 2, 5 (set bits 0, 2 and 5 of A)
UE2		Enable selected levels of group 2

NOTE

Execution of these instructions does not clear the interrupt structure and, therefore, does not affect any interrupt levels other than those selected. If a level selected for enabling is already enabled, it is not cleared by the execution of these instructions.

Cycles 1
Reference pages 2-9, 3-1, 3-73

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0063.01	Unitarily Inhibit group 1 interrupts	1 E/I	UI1
0063.02	group 2	2 E/I	UI2
0063.04	group 3	3 E/I	UI3



Any number of the 24 interrupt levels in the specified group are selectively inhibited; i.e., the selected bits of the Enable/Inhibit (E/I) register are reset to ZERO.

The corresponding bit(s) of register A must be set to select the appropriate level(s) prior to executing these instructions.

Example: Inhibit levels 1, 4 and 7 of group 3

TOA	B1B4B7	Select levels 1, 4, 7 (set bits 1, 4 and 7 of A)
UI3		Inhibit selected levels of group 3

NOTE

Execution of these instructions does not clear the interrupt structure and, therefore, does not affect any interrupt levels other than those selected. If one or more of the selected levels is active upon execution of these instructions, the level(s) will be placed in a "permissive" state.

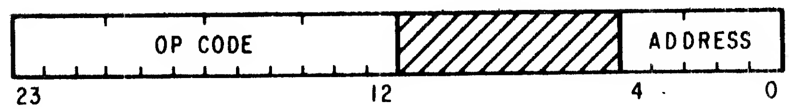
Cycles 1
Reference pages 2-9, 3-1, 3-73

This Page Left Blank Intentionally.

3-12 PROGRAM RESTRICT INSTRUCTIONS

The following instructions provide control for the optional Program Restrict system.

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0067:a	Branch and Link Unrestricted	J, P	BLU



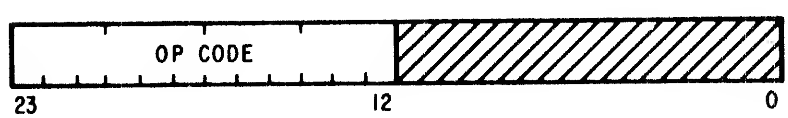
The next sequential address (PROGRAM ADDRESS +1) replaces the contents of register J and the contents of register P (current PROGRAM ADDRESS) are replaced by the 5-bit immediate memory address.

NOTE

> Execution of the BLU instruction will turn OFF the Program Restricted Flag (PRF). If the computer is in a HALT condition and the PRF is ON, the BLU instruction will be treated as a NOP instruction.

Cycles 1
Reference pages 2-12, 3-1

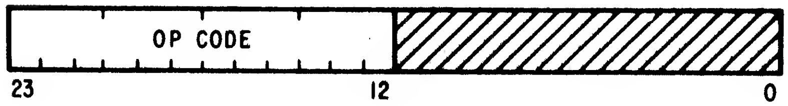
0056.	Transfer Double to Limit registers	LL, UL	TDL



The contents of bits E_0-E_{15} replace the previous contents of the Lower Limit (LL) register and the contents of bits A_0-A_{15} replace the previous contents of the Upper Limit (UL) register. Bits A_{21} and A_{22} set the restrict mode flags.

Cycles 1
Reference pages 2-12, 3-1

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0057.	Transfer Limit registers to Double	E,A	TLD



The contents of the Limit registers replace the previous contents of register D (E and A). The Upper Limit register contents are transferred to bits A_0-A_{15} and the contents of the Lower Limit register are transferred to E_0-E_{15}. The states of the restrict mode flags are transferred to bits A_{21} and A_{22}. All other bits in E and A are reset to ZERO.

Cycles 1
Reference pages 2-12, 3-1

This Page Left Blank Intentionally.

3-13 SHIFT INSTRUCTIONS

The Shift instruction group consists of arithmetic and logical shifts. The arithmetic shifts cause the contents of a register to be shifted left or right a specified number of times, while preserving the original sign. The logical shifts are similar to the arithmetic shifts, except that the sign bit is shifted along with the other bits.

With both types of shift instructions, any number of shifts from 1 to 255 may be programmed without restriction. The number of shifts (n) are specified in bits 0-7 of the instruction word.

At the conclusion of any shift operation, the Condition register is set to the status of the affected register's contents (Positive, Negative, Zero).

The time required to complete a Shift instruction is determined by the formula:

$$t = 2+[(n-1)/4]$$

where: t = time in cycles

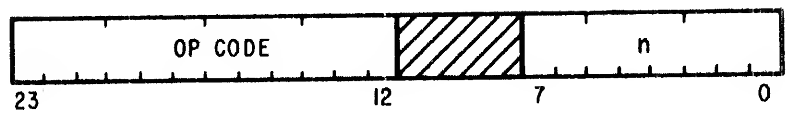
and

n = number of shifts

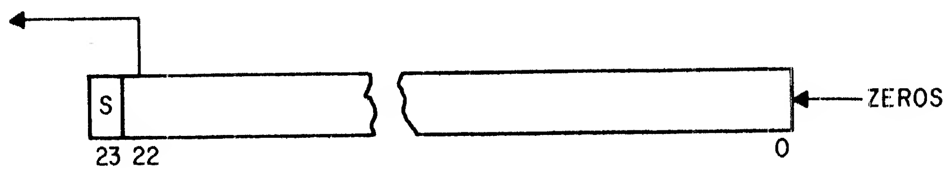
The following instructions are included in the Shift group.

MNEMONIC	INSTRUCTION	PAGE
LAA	Left shift Arithmetic A	3-90
LAD	Left shift Arithmetic Double	3-91
LLA	Left shift Logical A	3-92
LLD	Left shift Logical Double	3-92
LRA	Left Rotate A	3-93
LRD	Left Rotate Double	3-93
RAA	Right shift Arithmetic A	3-94
RAD	Right shift Arithmetic Double	3-94
RLA	Right shift Logical A	3-95
RLD	Right shift Logical Double	3-95
RRA	Right Rotate A	3-96
RRD	Right Rotate Double	3-96

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0040:n	Left shift Arithmetic A	A,C	LAA



Bits A_{22}-A_0 are shifted left n places, with the most significant n bits being lost and n ZEROs being shifted into the least significant bit positions. The sign bit (A_{23}) is unchanged.

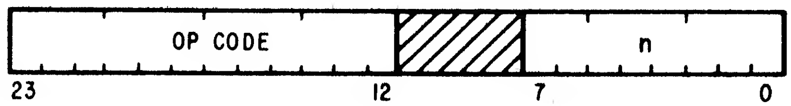


NOTE

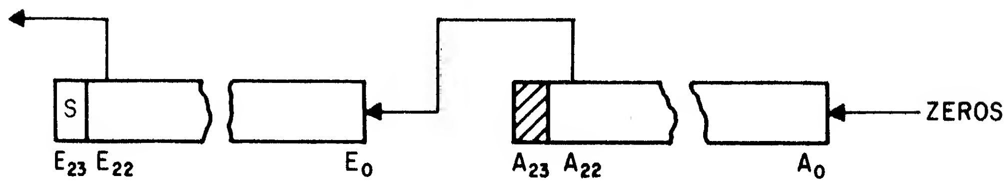
If a bit shifted off from A_{22} differs from the sign bit, the Condition register will be set to OVERFLOW. (This is in addition to the Positive/Negative/Zero status.)

Cycles .... 2 +[(n−1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0046:n	Left shift Arithmetic Double	E,A,C	LAD



Bits E_{22}-E_0 and A_{22}-A_0 are shifted —as one register— left n places. The most significant n bits are lost and the least significant n bits are replaced with ZEROs. Bits E_{23} and A_{23} are bypassed. E_{23} is the register D sign bit and A_{23} is not used in the double-precision format.

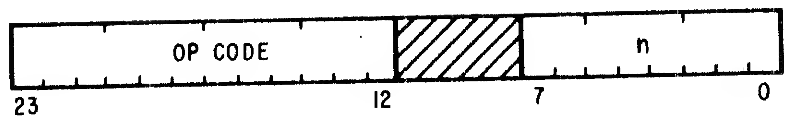


NOTE

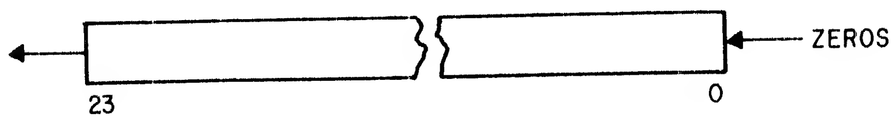
If a bit shifted off from E_{22} differs from the sign bit, the Condition register will be set to OVERFLOW. (This is in addition to the Positive/Negative/Zero status.)

Cycles .... 2 +[(n-1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0042:n	Left shift Logical A	A,C	LLA

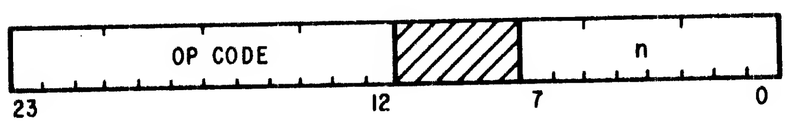


Bits A_{23}-A_0 are shifted left n places, with the most significant n bits being lost and the least significant n bits replaced by ZEROs.

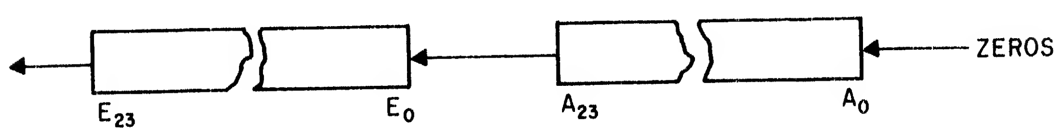


Cycles ....2+[(n-1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0050:n	Left shift Logical Double	E,A,C	LLD

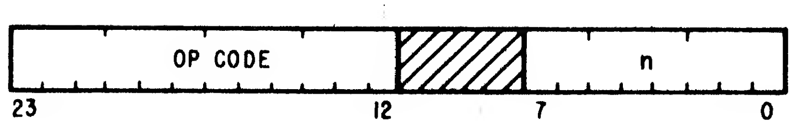


Bits E_{23}-E_0 and A_{23}-A_0 are shifted —as one register— left n places. The most significant n bits are lost and the least significant n bits are replaced with ZEROs.

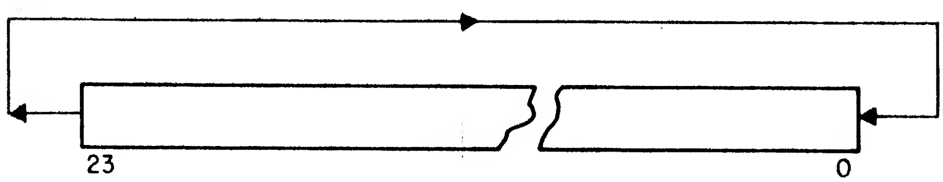


Cycles ....2+[(n-1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0044:n	Left Rotate A	A,C	LRA

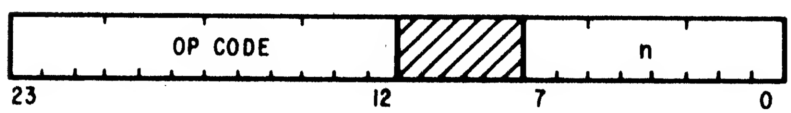


Bits A_{23}-A_0 are rotated left n places. No bits are lost.

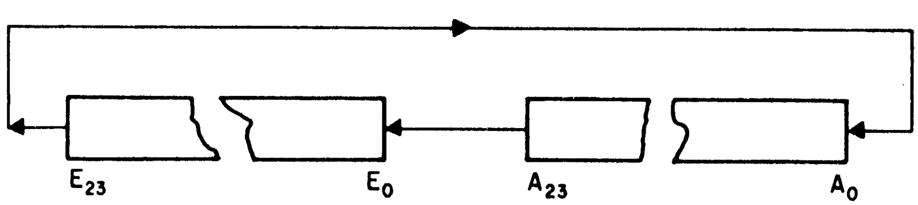


Cycles . . . . 2+[(n-1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0052:n	Left Rotate Double	E,A,C	LRD

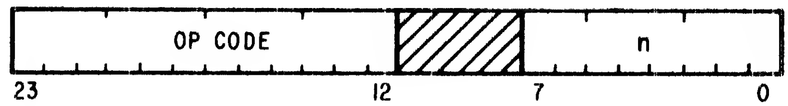


Bits E_{23}-E_0 and A_{23}-A_0 are rotated —as one register— left n places, with E_{23} replacing A_0 and A_{23} replacing E_0 as each shift takes place. No bits are lost.

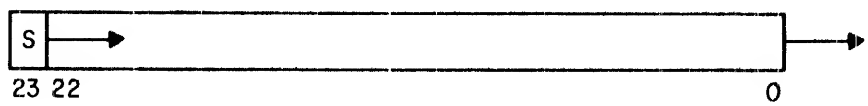


Cycles . . . . 2+[(n-1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0041:n	Right shift Arithmetic A	A,C	RAA

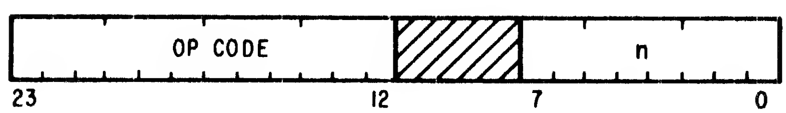


Bits A_{22}-A_0 are shifted right n places. The least significant n bits are lost and the most significant n bits are replaced by an extension of the sign bit (A_{23}). The sign bit is not changed.

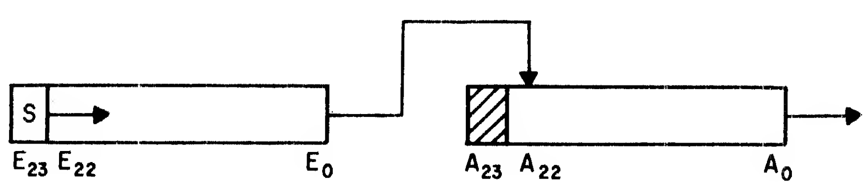


Cycles ....2+[(n-1)/4]
Reference pages 3-1, 3-89

0047:n	Right shift Arithmetic Double	E,A,C	RAD

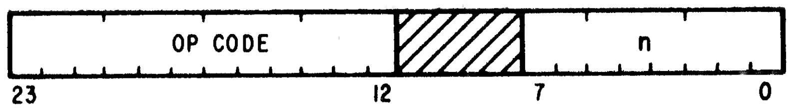


Bits E_{22}-E_0 and A_{22}-A_0 are shifted —as one register— right n places. The least significant n bits are lost and the most significant n bits are replaced by an extension of the sign bit (E_{23}). Bit A23 is bypassed.

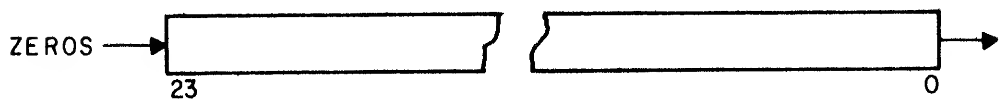


Cycles ....2+[(n-1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0043:n	Right shift Logical A	A,C	RLA

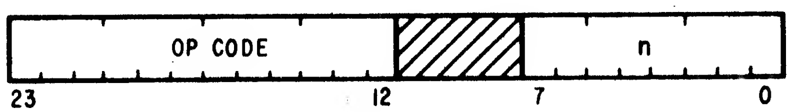


Bits A_{23}-A_0 are shifted right n places. The least significant n bits are lost and the most significant n bits are replaced by ZEROs.

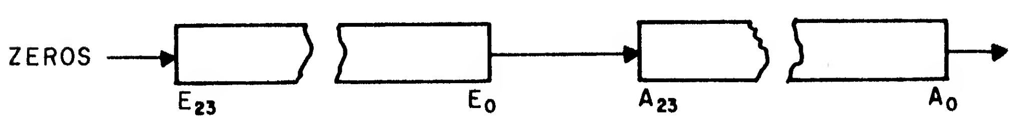


Cycles .... 2+[(n-1/4)]
Reference pages 3-1, 3-89

0051:n	Right shift Logical Double	E,A,C	RLD

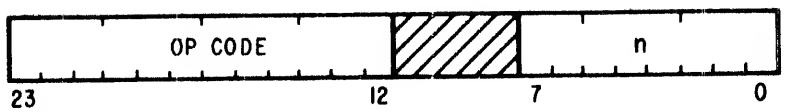


Bits E_{23}-E_0 and A_{23}-A_0 are shifted —as one register— right n places. The least significant n bits are lost and the most significant n bits are replaced by ZEROs.

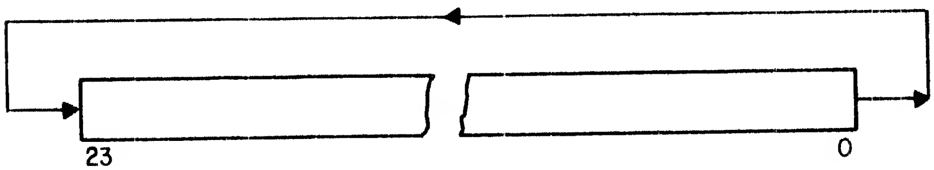


Cycles .... 2+[(n-1)/4]
Reference pages 3-1, 3-89

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0045:n	Right Rotate A	A,C	RRA

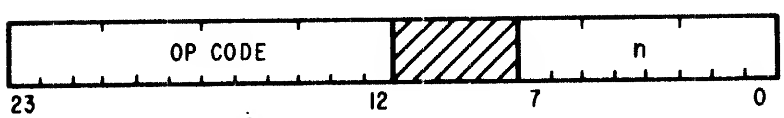


Bits A_{23}-A_0 are rotated right n places. No bits are lost.

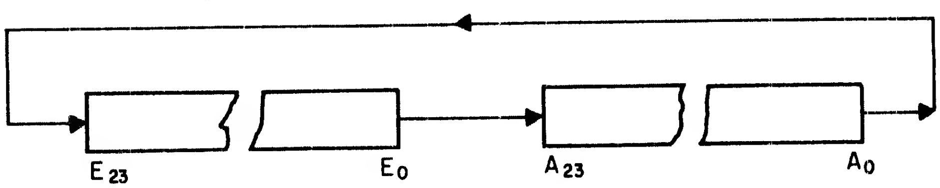


Cycles .... 2+[(n-1)/4]
Reference pages 3-1, 3-89

0053:n	Right Rotate Double	E,A,C	RRD



Bits E_{23}-E_0 and A_{23}-A_0 are rotated —as one register— right n places, with E_0 replacing A_{23} and A_0 replacing E_{23} as each shift takes place. No bits are lost.

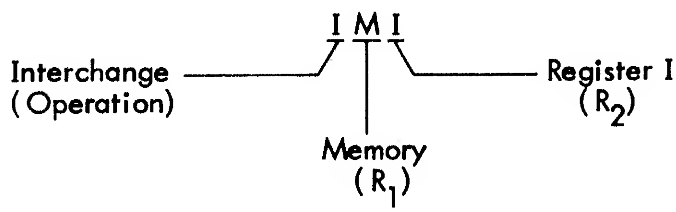


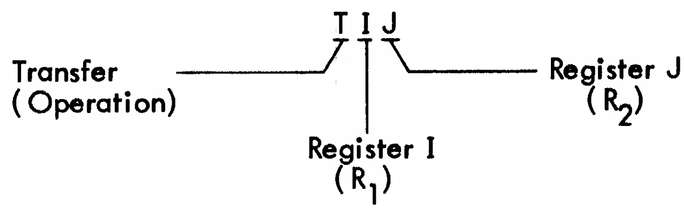
Cycles .... 2+[(n-1)/4]
Reference pages 3-1, 3-89

3-14 TRANSFER INSTRUCTIONS

The Transfer instruction group includes various types of operations. Among these are: interchanges between memory and a specified register, interchanges between registers, memory-to-register and register-to-memory transfers,and register-to-register transfers.

The mnemonic code for the Transfer instruction describes the individual operation. The first letter of the mnemonic indicates what action is to be taken; "I" for Interchange or "T" for Transfer. The second and third letters specify the source (R_1) and destination (R_2) respectively. Some examples are listed below:





With the exception of the Interchange instructions, the transfer source (R_1) is not altered by the execution of any instructions in the Transfer group.

The Condition register is always set to reflect the status (Positive, Negative or Zero) of the contents of R_2 at the completion of the instruction.

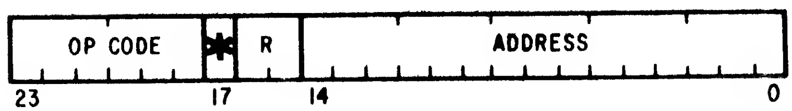
The following instructions are included in the Transfer group.

MNEMONIC	INSTRUCTION	PAGE
IAE	Interchange A and E	3-102
IAI	Interchange A and I	3-102
IAJ	Interchange A and J	3-102
IAK	Interchange A and K	3-102
IAT	Interchange A and T	3-102
IEA	Interchange E and A	3-102
IEI	Interchange E and I	3-102
IEJ	Interchange E and J	3-102
IEK	Interchange E and K	3-102
IET	Interchange E and T	3-102
IIA	Interchange I and A	3-102
IIE	Interchange I and E	3-102
IIJ	Interchange I and J	3-102
IIK	Interchange I and K	3-102
IIT	Interchange I and T	3-102
IJA	Interchange J and A	3-102

MNEMONIC	INSTRUCTION	PAGE
IJE	Interchange J and E	3-102
IJI	Interchange J and I	3-102
IJK	Interchange J and K	3-102
IJT	Interchange J and T	3-102
IKA	Interchange K and A	3-102
IKE	Interchange K and E	3-102
IKI	Interchange K and I	3-102
IKJ	Interchange K and J	3-102
IKT	Interchange K and T	3-102
IMA	Interchange M and A	3-101
IME	Interchange M and E	3-101
IMI	Interchange M and I	3-101
IMJ	Interchange M and J	3-101
IMK	Interchange M and K	3-101
ITA	Interchange T and A	3-102
ITE	Interchange T and E	3-102
ITI	Interchange T and I	3-102
ITJ	Interchange T and J	3-102
ITK	Interchange T and K	3-102
T1D	Transfer group 1 to Double	3-115
T2D	Transfer group 2 to Double	3-115
T3D	Transfer group 3 to Double	3-115
TAE	Transfer A to E	3-113
TAI	Transfer A to I	3-113
TAJ	Transfer A to J	3-113
TAK	Transfer A to K	3-113
TAM	Transfer A to Memory	3-112
TAT	Transfer A to T	3-113
TBM	Transfer Byte to Memory	3-110
TD1	Transfer Double to group 1	3-115
TD2	Transfer Double to group 2	3-115
TD3	Transfer Double to group 3	3-115
TDL	Transfer Double to Limit registers	3-114
TDM	Transfer Double to Memory	3-110
TEA	Transfer E to A	3-113
TEB	Transfer E to Byte	3-109
TEI	Transfer E to I	3-113
TEJ	Transfer E to J	3-113
TEK	Transfer E to K	3-113
TEM	Transfer E to Memory	3-112
TET	Transfer E to T	3-113
TFM	Transfer Flag to Memory	3-111
TIA	Transfer I to A	3-113
TIB	Transfer I to Byte	3-109
TIE	Transfer I to E	3-113
TIJ	Transfer I to J	3-113
TIK	Transfer I to K	3-113
TIM	Transfer I to Memory	3-112
TIT	Transfer I to T	3-113
TJA	Transfer J to A	3-113
TJB	Transfer J to Byte	3-109

MNEMONIC	INSTRUCTION	PAGE
TJE	Transfer J to E	3-113
TJI	Transfer J to I	3-113
TJK	Transfer J to K	3-113
TJM	Transfer J to Memory	3-112
TJT	Transfer J to T	3-113
TKA	Transfer K to A	3-113
TKB	Transfer K to Byte	3-109
TKE	Transfer K to E	3-113
TKI	Transfer K to I	3-113
TKJ	Transfer K to J	3-113
TKM	Transfer K to Memory	3-112
TKT	Transfer K to T	3-113
TLD	Transfer Limit registers to Double	3-114
TLO	Transfer Long Operand to K	3-108
TMA	Transfer Memory to A	3-105
TMB	Transfer Memory to Byte	3-103
TMD	Transfer Memory to Double	3-103
TME	Transfer Memory to E	3-105
TMI	Transfer Memory to I	3-105
TMJ	Transfer Memory to J	3-105
TMK	Transfer Memory to K	3-105
TMQ	Transfer Memory to Query register	3-104
TMR	Transfer Memory to Registers	3-105
TNA	Transfer Negative operand to A	3-106
TNE	Transfer Negative operand to E	3-106
TNI	Transfer Negative operand to I	3-106
TNJ	Transfer Negative operand to J	3-106
TNK	Transfer Negative operand to K	3-106
TNT	Transfer Negative operand to T	3-106
TOA	Transfer Operand to A	3-107
TOB	Transfer Operand to Byte	3-106
TOC	Transfer Operand to Condition Register	3-107
TOE	Transfer Operand to E	3-107
TOI	Transfer Operand to I	3-107
TOJ	Transfer Operand to J	3-107
TOK	Transfer Operand to K	3-107
TOT	Transfer Operand to T	3-107
TRM	Transfer Registers to Memory	3-112
TSA	Transfer Switches to A	3-108
TSE	Transfer Switches to E	3-108
TSI	Transfer Switches to I	3-108
TSJ	Transfer Switches to J	3-108
TSK	Transfer Switches to K	3-108
TST	Transfer Switches to T	3-108
TTA	Transfer T to A	3-113
TTB	Transfer T to Byte	3-109
TTE	Transfer T to E	3-113
TTI	Transfer T to I	3-113
TTJ	Transfer T to J	3-113
TTK	Transfer T to K	3-113
TZA	Transfer Zero to A	3-109

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
66. *+1:a	Interchange Memory and I	M,I,C	IMI
66. *+2:a	J	M,J,C	IMJ
66. *+3:a	K	M,K,C	IMK



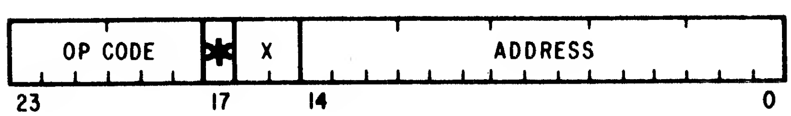
The contents of the effective memory address and register I, J, or K are interchanged.

NOTE

The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

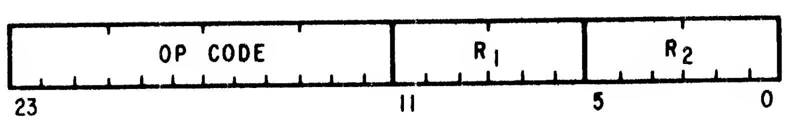
67. *+X:a	Interchange Memory and E	M,E,C	IME
70. *+X:a	A	M,A,C	IMA



The contents of the effective memory address and the specified register are interchanged.

Cycles .... 3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

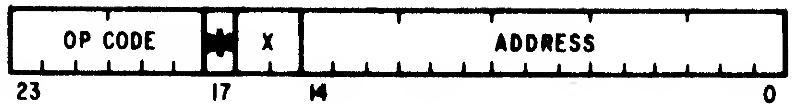
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0035.01.02	Interchange I and J	I,J,C	IIJ
0035.01.04	K	I,K,C	IIK
0035.01.10	E	I,E,C	IIE
0035.01.20	A	I,A,C	IIA
0035.01.40	T	I,T,C	IIT
0035.02.01	Interchange J and I	J,I,C	IJI
0035.02.04	K	J,K,C	IJK
0035.02.10	E	J,E,C	IJE
0035.02.20	A	J,A,C	IJA
0035.02.40	T	J,T,C	IJT
0035.04.01	Interchange K and I	K,T,C	IKI
0035.04.02	J	K,J,C	IKJ
0035.04.10	E	K,E,C	IKE
0035.04.20	A	K,A,C	IKA
0035.04.40	T	K,T,C	IKT
0035.10.01	Interchange E and I	E,I,C	IEI
0035.10.02	J	E,J,C	IEJ
0035.10.04	K	E,K,C	IEK
0035.10.20	A	E,A,C	IEA
0035.10.40	T	E,T,C	IET
0035.20.01	Interchange A and I	A,I,C	IAI
0035.20.02	J	A,J,C	IAJ
0035.20.04	K	A,K,C	IAK
0035.20.10	E	A,E,C	IAE
0035.20.40	T	A,T,C	IAT
0035.40.01	Interchange T and I	T,I,C	ITI
0035.40.02	J	T,J,C	ITJ
0035.40.04	K	T,K,C	ITK
0035.40.10	E	T,E,C	ITE
0035.40.20	A	T,A,C	ITA



The contents of R_1 and R_2 are interchanged.

Cycles 1
Reference pages 3-1, 3-97

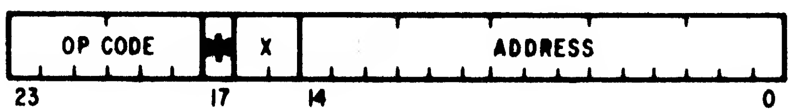
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
07. *+X:a	Transfer Memory to Byte	A, C	TMB



The 8 least significant bits (0-7) of the contents of the effective memory address replace the previous contents of register B (A_0-A_7). Bits A_8-A_{23} are unaffected.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

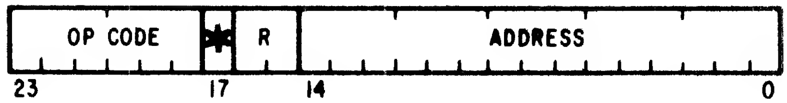
06. *+X:a	Transfer Memory to Double	E, A, C	TMD



The contents of the effective memory address (EMA) and the next sequential address (EMA +1) replace the previous contents of register D (E and A). EMA and EMA +1 are transferred to E and A, respectively.

Cycles ....3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
51. *+0:a	Transfer Memory to Query register	Q,C	TMQ



Bits 0-15 of the contents of the effective memory address replace the previous contents of the Query register.

Executing this instruction will cause the Address Trap to be enabled or disabled, depending on the state of bit 23 of the effective memory address.

Bit 23 = ONE = Disable Address Trap
Bit 23 = ZERO = Enable Address Trap

Example:

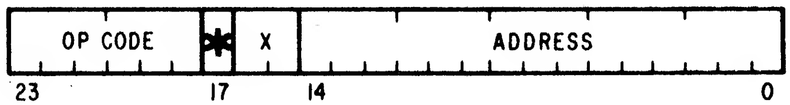
```
        TMQ    OA
        . . .
OA      DAC    ADDR      Enable Address Trap

         or

OA      DAC*   O         Disable Address Trap
```

NOTE

The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles . . . . 2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

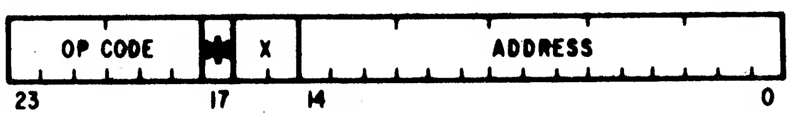
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
01. *+X:a	Transfer Memory to I	I,C	TMI
02. *+X:a	J	J,C	TMJ
03. *+X:a	K	K,C	TMK
04. *+X:a	E	E,C	TME
05. *+X:a	A	A,C	TMA



The contents of the effective memory address replace the previous contents of the specified register.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

10. *+X:a	Transfer Memory to Registers	I,J,K,E,A	TMR



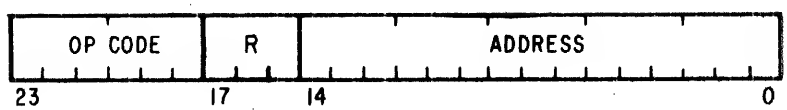
Registers I, J, K, E and A are loaded from consecutive memory addresses beginning with the effective memory address.

NOTE

External interrupts are prohibited for the period of one instruction following the execution of this instruction.

Cycles ....6 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

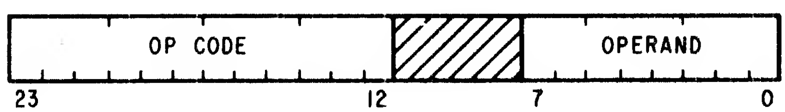
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
63. 1:o	Transfer Negative operand to I	I, C	TNI
63. 2:o	J	J, C	TNJ
63. 3:o	K	K, C	TNK
63. 4:o	E	E, C	TNE
63. 5:o	A	A, C	TNA
63. 6:o	T	T, C	TNT



The two's complement of the 15-bit unsigned operand replaces the previous contents of bits 0-23 of the specified register.

Cycles 1
Reference pages 3-1, 3-2, 3-97

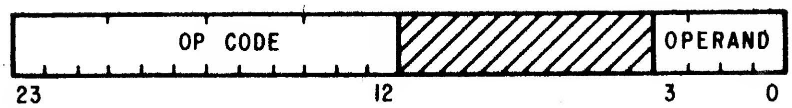
0003:o	Transfer Operand to Byte	A, C	TOB



The 8-bit operand replaces the previous contents of register B (A_0-A_7). Bits A_8-A_{23} are unaffected.

Cycles 1
Reference pages 3-1, 3-97

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0036:o	Transfer Operand to Condition register	C	TOC



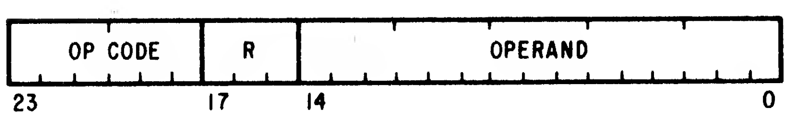
The 4-bit operand replaces the previous contents of the Condition register.

Operand definition is as follows:

bit 0 - ONE = Overflow,	ZERO = No Overflow
bit 1 - ONE = Negative,	ZERO = Not Negative
bit 2 - ONE = Zero,	ZERO = Not Zero
bit 3 - ONE = Positive,	ZERO = Not Positive

Cycles 1
Reference pages 3-1, 3-97

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
62. 1:o	Transfer Operand to I	I, C	TOI
62. 2:o	J	J, C	TOJ
62. 3:o	K	K, C	TOK
62. 4:o	E	E, C	TOE
62. 5:o	A	A, C	TOA
62. 6:o	T	T, C	TOT



The 15-bit operand replaces the previous contents of bits 0-23 of the specified register.

Cycles 1
Reference pages 3-1, 3-97

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
23.6:A	Transfer Long Operand to K	K	TLO

OP CODE	I	OPERAND OR ADDRESS
23	17	15 ... 0

The 16-bit operand (or address) replaces the previous contents of bits 0-23 of register K.

Cycles 1
Reference pages 3-1, 3-2, 3-97

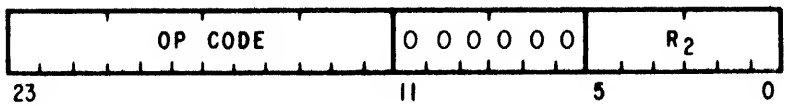
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0031.00.01	Transfer Switches to I	I,C	TSI
0031.00.02	J	J,C	TSJ
0031.00.04	K	K,C	TSK
0031.00.10	E	E,C	TSE
0031.00.20	A	A,C	TSA
0031.00.40	T	T,C	TST

OP CODE	R_1	R_2
23	11	5 ... 0

The states (set = ONE) of the console control switches (i.e., switch register) are transferred to the corresponding bit positions of the specified register.

Cycles 1
Reference pages 3-1, 3-97

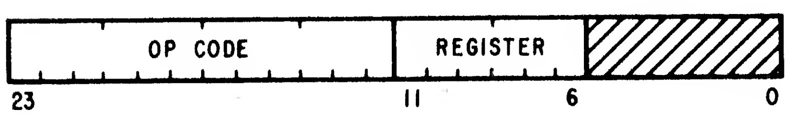
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0030.00.01	Transfer Zero to I	I,C	TZI
0030.00.02	J	J,C	TZJ
0030.00.04	K	K,C	TZK
0030.00.10	E	E,C	TZE
0030.00.20	A	A,C	TZA
0030.00.40	T	T,C	TZT
0030.00.30	Double	E,A,C	TZD



The previous contents of the specified register are replaced with ZEROs.

Cycles 1
Reference pages 3-1, 3-97

0002.01	Transfer I to Byte	A	TIB
0002.02	J	A	TJB
0002.04	K	A	TKB
0002.10	E	A	TEB
0002.40	T	A	TTB



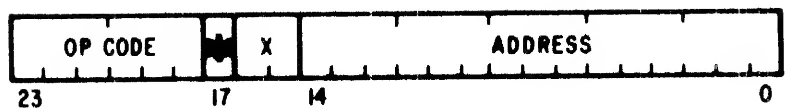
The least significant 8 bits (0-7) of the contents of the specified register replace the previous contents of register B (A_0-A_7). Bits A_8-A_{23} are unchanged.

NOTE

The Condition register is not affected.

Cycles 1
Reference pages 3-1, 3-97

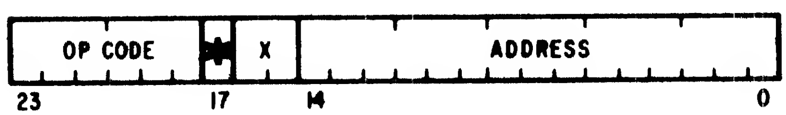
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
17. *+X:a	Transfer Byte to Memory	M	TBM



The contents of register B (A_0-A_7) replace the 8 least significant bits (0-7) of the contents of the effective memory address. Bits 8-23 of the memory word are unaffected.

Cycles ....3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

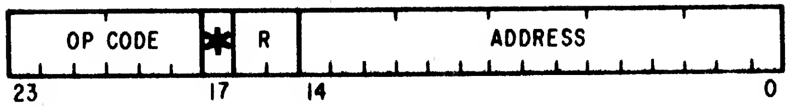
16. *+X:a	Transfer Double to Memory	M	TDM



The contents of register D (E and A) replace the previous contents of the effective memory address (EMA) and the next sequential address (EMA +1). The contents of E and A are transferred to EMA and EMA +1, respectively.

Cycles ....3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
46. *+0:a	Transfer Flag to Memory	M,C	TFM



The previous contents of the effective memory address are replaced by ONEs.

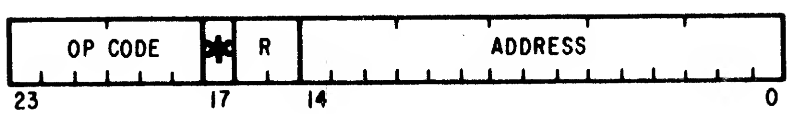
NOTES

(1) The Condition (C) register is set to the status of memory <u>prior</u> to the transfer.

(2) The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

66. *+0:a	Transfer Zero to Memory	M,C	TZM



The previous contents of the effective memory address are replaced by ZEROs.

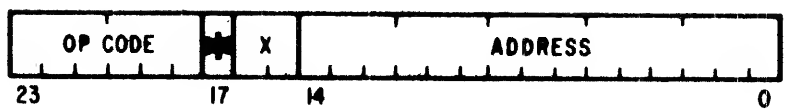
NOTES

(1) The Condition (C) register is set to the status of memory <u>prior</u> to the transfer.

(2) The immediate memory reference cannot be indexed; however, indexing of indirect references is permitted.

Cycles .... 3 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

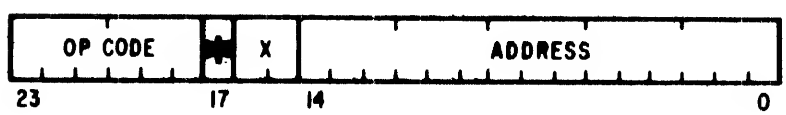
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
11. *+X:a	Transfer I to Memory	M	TIM
12. *+X:a	J	M	TJM
13. *+X:a	K	M	TKM
14. *+X:a	E	M	TEM
15. *+X:a	A	M	TAM



The contents of the specified register replace the previous contents of the effective memory address.

Cycles ....2 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

20. *+X:a	Transfer Registers to Memory	M	TRM



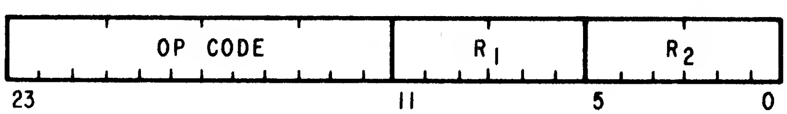
The contents of registers I, J, K, E and A are stored in consecutive memory locations beginning with the effective memory address.

NOTE

External interrupts are prohibited for the period of one instruction following the execution of this instruction.

Cycles ....6 (+1 per indirect reference)
Reference pages 3-1, 3-2, 3-97

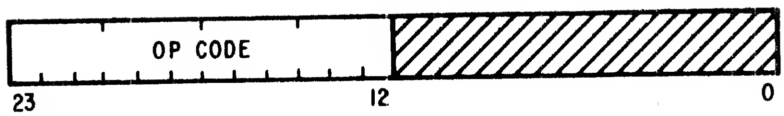
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0030.01.02	Transfer I to J	J, C	TIJ
0030.01.04	K	K, C	TIK
0030.01.10	E	E, C	TIE
0030.01.20	A	A, C	TIA
0030.01.40	T	T, C	TIT
0030.02.01	Transfer J to I	I, C	TJI
0030.02.04	K	K, C	TJK
0030.02.10	E	E, C	TJE
0030.02.20	A	A, C	TJA
0030.02.40	T	T, C	TJT
0030.04.01	Transfer K to I	I, C	TKI
0030.04.02	J	J, C	TKJ
0030.04.10	E	E, C	TKE
0030.04.20	A	A, C	TKA
0030.04.40	T	T, C	TKT
0030.10.01	Transfer E to I	I, C	TEI
0030.10.02	J	J, C	TEJ
0030.10.04	K	K, C	TEK
0030.10.20	A	A, C	TEA
0030.10.40	T	T, C	TET
0030.20.01	Transfer A to I	I, C	TAI
0030.20.02	J	J, C	TAJ
0030.20.04	K	K, C	TAK
0030.20.10	E	E, C	TAE
0030.20.40	T	T, C	TAT
0030.40.01	Transfer T to I	I, C	TTI
0030.40.02	J	J, C	TTJ
0030.40.04	K	K, C	TTK
0030.40.10	E	E, C	TTE
0030.40.20	A	A, C	TTA



The contents of R_1 replace the previous contents of R_2.

Cycles 1
Reference pages 3-1, 3-97

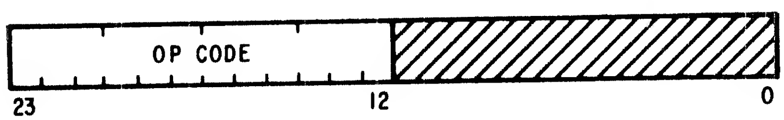
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0056.	Transfer Double to Limit registers	LL,UL	TDL



The contents of bits E_0-E_{15} replace the previous contents of the Lower Limit (LL) register and the contents of bits A_0-A_{15} replace the previous contents of the Upper Limit (UL) register. Bits A_{21} and A_{22} set the restrict mode flags.

Cycles 1
Reference pages 2-12, 3-1, 3-97

0057.	Transfer Limit registers to Double	E,A	TLD



The contents of the Limit registers replace the previous contents of register D (E and A). The Upper Limit register contents are transferred to bits A_0-A_{15} and the contents of the Lower Limit register are transferred to E_0-E_{15}. The states of the restrict mode flags are transferred to bits A_{21} and A_{22}. All other bits in E and A are reset to ZERO.

Cycles 1
Reference pages 2-12, 3-1, 3-97

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0064.01	Transfer Double to group 1	1 A/D, 1 E/I	TD1
0064.02	group 2	2 A/D, 2 E/I	TD2
0064.04	group 3	3 A/D, 3 E/I	TD3

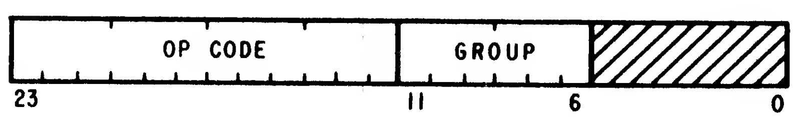
The contents of register D (E and A) replace the previous contents of the Arm/Disarm (A/D) and Enable/Inhibit (E/I) registers of the specified interrupt group (1, 2 or 3). The contents of E are transferred to the A/D register and the contents of A are transferred to the E/I register.

NOTE

The external interrupt structure is cleared by the execution of these instructions.

Cycles 1
Reference pages 2-11, 3-1, 3-73, 3-97

0065.01	Transfer group 1 to Double	E,A	T1D
0065.02	group 2	E,A	T2D
0065.04	group 3	E,A	T3D



The contents of the Arm/Disarm (A/D) and Enable/Inhibit (E/I) registers of the specified interrupt group replace the previous contents of register D (E and A). The contents of the A/D register are transferred to register E and the contents of the E/I register are transferred to register A.

NOTE

The states of the external interrupts are not affected by the execution of these instructions.

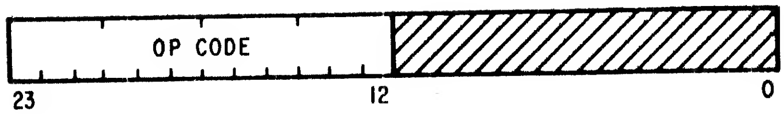
Cycles 1
Reference pages 2-11, 3-1, 3-73, 3-97

This Page Left Blank Intentionally.

3-15 MISCELLANEOUS INSTRUCTIONS

The following instructions are included in the Miscellaneous group because they do not fall into one of the previously defined functional groups.

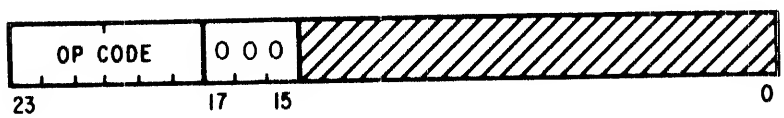
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0000.	HaLT	-	HLT



The PROGRAM ADDRESS (i.e., the contents of the P register) is advanced by one and program execution is terminated. When the RUN switch is depressed, execution will begin at the location defined by the PROGRAM ADDRESS.

Cycles 1
Reference page 3-1

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
62.0	No OPeration	-	NOP



The PROGRAM ADDRESS is advanced by one and program execution continues with the next instruction.

Cycles 1
Reference page. 3-1

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
24.4:o	Generate Argument Pointer	I,J	GAP

OP CODE	I 0 0	OPERAND
23	17 15	14 0

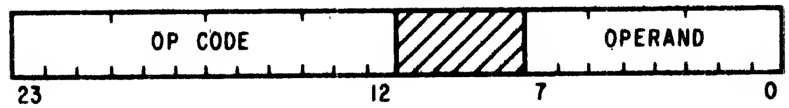
The contents of register J are assumed to be the first address in an indirect memory reference sequence. The effective memory address derived from this indirect sequence replaces the previous contents of register I. The contents of register J and the 15-bit operand are added, and the result is placed in register J.

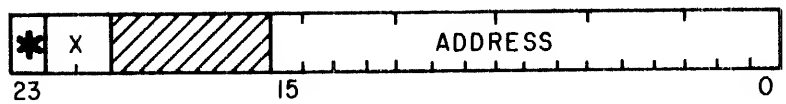
Example:

```
Y     DATA   2
      ...
X     DAC    Y
      ...
A     BLJ    B        (J) = C, (P) = B
C     DAC*   X
D     ...             Return
B     GAP    1        (I) = Y, (J) = (J)+1
      TMA    0,I      (A) = 2
      BUC    0,J      (P) = D
```

Cycles ....2 (+1 per indirect reference)
Reference page 3-1

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0055:o *+X:A	Update Stack Pointer	K,C	USP





The contents of register K are replaced by the contents of the effective memory address. The 8-bit signed operand is then added to the effective memory address.

Example:

	BLJ	ENT	Call re-entrant routine
	...		
ENT	TRM*	SP	Save registers in stack
	USP	5	Update Stack Pointer [(K) = stack, (SP)=stack+5]
	DAC	SP	
	...		
	HTK	SP	Reset stack pointer
	TMR*	SP	Restore registers
	BUC	0,J	Return
SP	DAC	STACK	Stack pointer
STACK	BLOK	5N	Where N represents maximum number of re-entrant levels

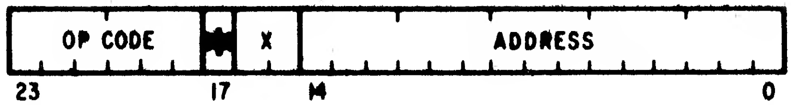
NOTES

(1) The Condition register is set to reflect the result of the operand addition.

(2) External interrupts are prohibited for the period of one instruction following this instruction.

Cycles ....4 (+1 per indirect reference)
Reference pages 3-1, 3-2

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
40. *+X:a	EXecute Memory	See Note 1	EXM



The instruction located in the effective memory address is executed as though it were at the address of the EXM.

In the case that the referenced instruction is a two word instruction, the second word must follow the EXM.

Example:

```
        EXM    M
        DAC    L       Second word
        . . .
M       AOM    10      Two word instruction
        AOM    20
        AOM    30
```

NOTES

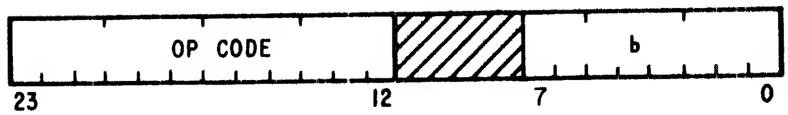
(1) The registers affected will depend on the instruction in the effective memory address.

(2) External interrupts are prohibited for the period of one instruction following the execution of this instruction.

Cycles . . . . 1 (+1 per indirect reference) plus the time required to execute the referenced instruction

Reference pages 3-1, 3-2

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0011:b	Query Bits of Byte	C	QBB



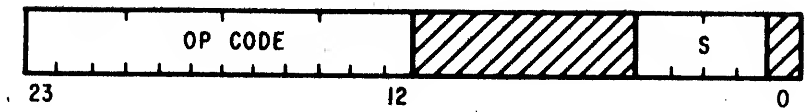
A logical AND is performed between operand bits 0-7 and the contents of register B. The Condition register is set according to the status of the result; i.e., positive, negative, or zero.

Examples:

1)	TOA	B7	A = ' 00000200	C = Positive
	. . .			
	. . .			
	QBB	B7		C = Negative
2)	TOA	B6	A = ' 00000100	C = Positive
	. . .			
	. . .			
	QBB	B6		C = Positive
3)	TNA	1	A = ' 77777777	C = Negative
	. . .			
	. . .			
	DMA	MASK	A = ' 40000000	C = Negative
	. . .			
	. . .			
MASK	DATA	' 40000000		

Cycles 1
Reference page 3-1

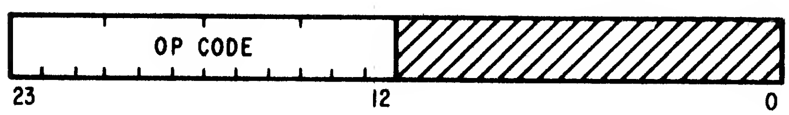
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0001:s	Query Sense Switches	C	QSS



A logical AND is performed between operand bits 1–4 and the state(s) of the sense switches. The Condition register is set accordingly.

Cycles 1
Reference page 3-1

0007.	Extend Zeros from Byte	A	EZB



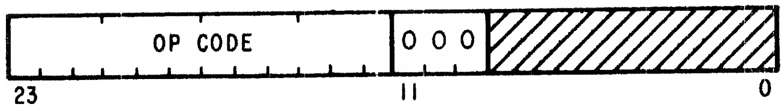
Bit positions A_8-A_{23} are set to ZERO. The contents of register B (A_0-A_7) are not affected.

NOTE

The Condition register is not affected.

Cycles 1
Reference page 3-1

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0077.0	Hold Interval Timer	-	HIT



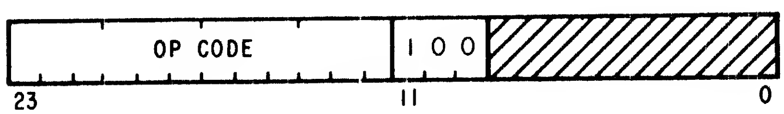
The CPU's Interval Timer is halted and will remain so until released by an RPT or RCT instruction.

NOTE

The 10-cycle counter associated with the Interval Timer is reset to zero by this instruction.

Cycles 1
Reference page 3-1

0077.4	Release Processor Time	-	RPT



The CPU's Interval Timer is started; i.e., allowed to begin its counting sequence. The Processor Time mode allows the Interval Timer to count only CPU time and not time stolen by an ABC operation.

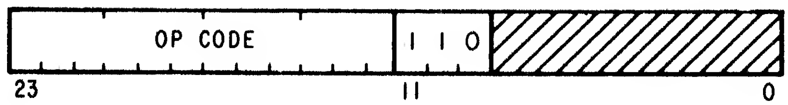
Once started, the timer counts until held by an HIT instruction or until the CPU is halted. At each 10-cycle interval, the contents of register T are decremented by one and tested for zero. If the contents of T are zero, an executive interrupt is triggered.

NOTE

The timer is not stopped by the executive interrupt.

Cycles 1
Reference page 3-1

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
0077.6	Release Clock Time	-	RCT



The CPU's Interval Timer is started; i.e., allowed to beain its counting sequence. The Clock Time mode causes the Interval Timer to count continuously.

Once started, the timer will count until held by an HIT instruction. At each 10-cycle interval, the contents of register T are decremented by one and tested for zero. If the contents of T are zero, an executive interrupt is triggered.

NOTE

The timer is not stopped by the interrupt.

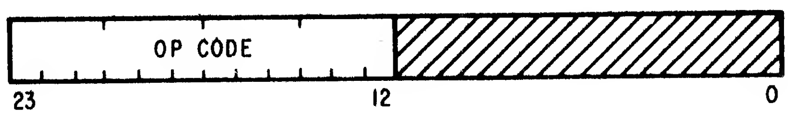
Cycles 1
Reference page 3-1

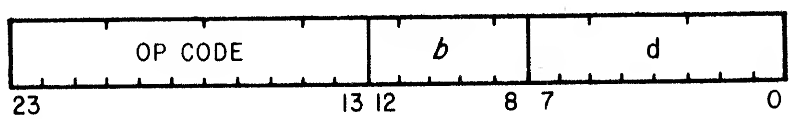
This Page Left Blank Intentionally.

3-16 BIT PROCESSOR INSTRUCTIONS

The bit (Boolean function) processor group of instructions include branches, logical manipulation, and interrogations of a specified bit selected from an effective memory address or the H register. In most instances, the Condition register will be set to reflect the status of the operation after completion of the instruction.

The bit processor employs two instruction word formats. The first format uses a (bits 12-24) OP CODE to specify the operation to be performed. The remaining 12 bits (bits 0-11) are undefined. The second instruction format contains a displacement, bit specification, and an OP CODE. Eight bits (bits 0-7) are added to the base address contained in register V to obtain a displacement from the base address which is an effective memory address for the word containing the bit in question. Five bits (bits 8-12) are used to select a specific bit in the effective memory address for an operation as specified in the 11-bit (bits 13-23) OP CODE. Both instruction word formats are illustrated below.

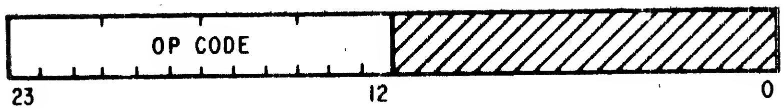




The following instructions are included in the Bit Processor group.

MNEMONIC	INSTRUCTION	PAGE
DMH	Dot Memory with H	3-131
DNH	Dot Not (memory) with H	3-131
FBM	Flag Bit of Memory	3-135
NHH	Negate of H to H	3-130
OMH	Or Memory with H	3-132
ONH	Or Not (memory) with H	3-132
QBH	Query bit of H	3-130
QBM	Query bit of Memory	3-134
TFH	Transfer Flag to H	3-128
THM	Transfer H to Memory	3-135
TKV	Transfer K to V	3-129
TMH	Transfer Memory to H	3-134
TVK	Transfer V to K	3-129
TZH	Transfer Zero to H	3-128
XMH	eXclusive-or Memory with H	3-133
XNH	eXclusive-or Not (memory) with H	3-133
ZBM	Zero Bit of Memory	3-136

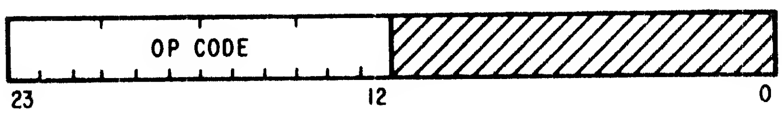
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7742.	Transfer Zero to H	H,C	TZH



A ZERO is placed in register H. The Condition register is set to reflect the original contents of H.

Cycles 1
Reference pages 3-1, 3-2, 3-127

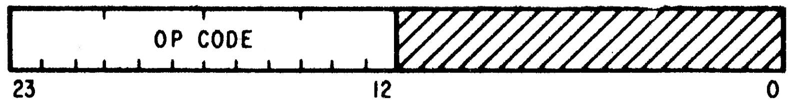
7743.	Transfer Flag to H	H,C	TFH



A ONE is placed in register H and the Condition register is set to reflect the previous contents of H.

Cycles 1
Reference pages 3-1, 3-2, 3-127

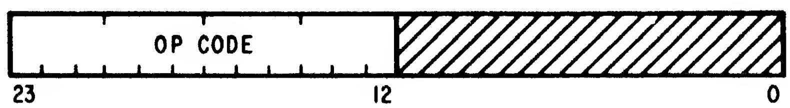
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7744.	Transfer K to V	V	TKV



The 16 least significant bits of register K replace the present contents of register V. The Condition register is unaffected.

Cycles 1
Reference pages 3-1, 3-2, 3-127

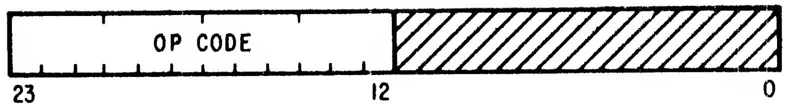
7745.	Transfer V to K	K	TVK



The contents of register V are transferred to the 16 least significant bit positions of register K. The Condition register is unaffected.

Cycles 1
Reference pages 3-1, 3-2, 3-127

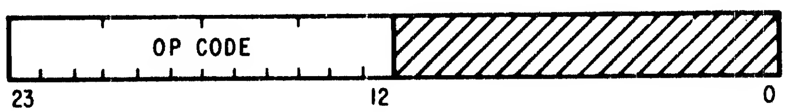
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7746.	Query Bit of H	C	QBH



The H register bit is tested and the Condition register is set to display the results of the query.

Cycles 1
Reference pages 3-1, 3-2, 3-127

7747.	Negate of H to H	H,C	NHH



The current content of register H is complemented and returned to H. The Condition register is set to display the result.

Cycles 1
Reference pages 3-1, 3-2, 3-127

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7750.*b*:d	Dot Memory with H	H,C	DMH

OP CODE	*b*	d
23 13	7	0

A logical AND is performed between the selected bit in the effective memory address and the contents of register H. The result is returned to the H register and the Condition register is set to display the result.

Cycles 2
Reference pages 3-1, 3-2, 3-127

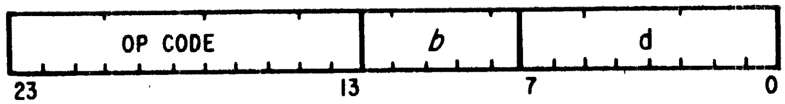
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7752.*b*:d	Dot Not (memory) with H	H,C	DNH

OP CODE	*b*	d
23 13	7	0

A logical AND is performed between the complement of the selected bit in the effective memory address and the content of register H. The result is returned to the H register and the Condition register is set to display the result.

Cycles 2
Reference pages 3-1, 3-2, 3-127

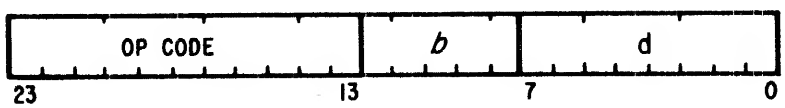
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7754.*b*:d	Or Memory with H	H,C	OMH



A logical OR is performed between the selected bit in the effective memory address and the content of register H. The Condition register is set to display the result.

Cycles 2
Reference pages 3-1, 3-2, 3-127

7756.*b*:d	Or Not (memory) with H	H,C	ONH



A logical OR is performed between the complement of the selected bit in the effective memory address and the content of register H. The Condition register is set to display the result.

Cycles 2
Reference pages 3-1, 3-2, 3-127

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7760.*b*:d	eXclusive-or Memory with H	H,C	XMH

OP CODE	*b*	d
23	13	7 0

An EXCLUSIVE-OR function is performed between the selected bit in the effective memory address and the content of register H. The Condition register is set to display the result.

Cycles 2
Reference pages 3-1, 3-2, 3-127

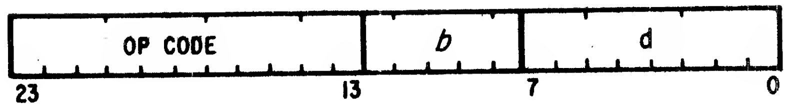
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7762.*b*:d	eXclusive-or Not (memory) with H	H,C	XNH

OP CODE	*b*	d
23	13	7 0

An EXCLUSIVE-OR function is performed between the complement of the selected bit in the effective memory address and the content of H register. The Condition register is set to display the result.

Cycles 2
Reference pages 3-1, 3-2, 3-127

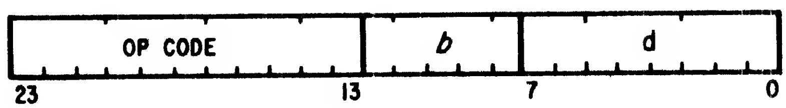
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7764.*b*:d	Transfer Memory to H	H, C	TMH



The selected bit in the effective memory address is transferred to register H. The Condition register is set to display the current content of the H register.

Cycles 2
Reference pages 3-1, 3-2, 3-127

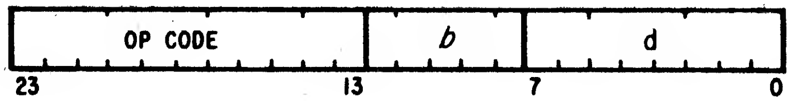
7766.*b*:d	Query Bit of Memory	C	QBM



The selected bit in the effective memory address is tested and the Condition register is set to display the result of the query.

Cycles 2
Reference pages 3-1, 3-2, 3-127

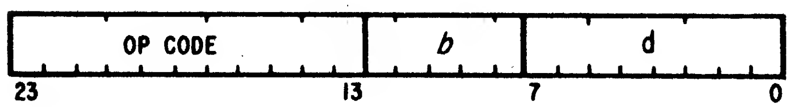
INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7770.*b*:d	Transfer H to Memory	M	THM



The content of register H is placed in the selected bit position in the effective memory address. The Condition register is not affected.

Cycles 3
Reference pages 3-1, 3-2, 3-127

7772.*b*:d	Flag Bit of Memory	M,C	FBM



A ONE is placed in the selected bit position in the effective memory address. The Condition register is set to display the original state of the selected bit in memory.

Cycles 3
Reference pages 3-1, 3-2, 3-127

INSTRUCTION FORMULA	FUNCTION	REGISTERS AFFECTED	MNEMONIC
7774.*b*:d	Zero Bit of Memory	M, C	ZBM

OP CODE	*b*	d
23 – 13	7	0

A ZERO is transferred to the selected bit position in the effective memory address. The Condition register is set to display the original state of the selected bit in memory.

Cycles 3
Reference pages 3-1, 3-2, 3-127

INSTRUCTION SET

SECTION IV

INPUT/OUTPUT

4-1 INTRODUCTION

The Series 6000 Computer input/output (I/O) structure combines the characteristic economy of unit I/O systems with the speed of a channel I/O system. This configuration, in conjunction with the I/O instruction repertoire, permits maximum flexibility in I/O communications.

The basic I/O structure allows single word data transfers between the Central Processing Unit (CPU) and a peripheral unit. It also allows I/O command and test operations to be program-controlled. An optional Automatic Block Controller (ABC) controls transfer of blocks of data between the CPU and the peripheral units without program intervention.

The relationship between the CPU and the I/O structure is illustrated on Figure 4-1. The elements comprising the I/O structure are described in the following paragraphs.

4-2 BASIC INPUT/OUTPUT STRUCTURE

The basic I/O structure involves communication between the CPU and a peripheral unit by way of a channel. The CPU communicates with a channel and the channel, in turn, communicates with a peripheral unit.

The I/O structure accommodates up to 28 channels for the DC 6024 and up to 14 channels for the DC 6024/3, all of which can be active concurrently. Each channel can communicate with from 1 to 16 peripheral units using the standard I/O instructions. Only one peripheral unit per channel may be active at any given time.

4-3 AUTOMATIC BLOCK CONTROLLER

The optional Automatic Block Controller (ABC) permits a block of data (n words) to be transferred between the CPU and peripheral units, without program intervention once activated. Data blocks of from one to 65,536 words may be automatically transferred. In addition, separate data blocks may be linked and transferred under control of the ABC.

Up to 14 ABCs may be used with the I/O structure. Each ABC services a single channel. Memory access by one or more ABCs is granted on a priority basis by the Automatic Block Control Director (ABCD). Two or more ABCs may be interlaced depending on the transfer capability of the peripheral units involved. A single ABC can transfer up to 833,333 words per second (2.5 million bytes). Multiple ABCs operating concurrently can transfer a maximum of 1,666,666 words or 5 million bytes per second. The above transfer rates are applicable to an ABC channel operating in conjunction with a DC 6024 Computer. The DC 6024/3 computer transfers data via a single channel at a rate up to 500,000 words per second (1.5 million bytes).

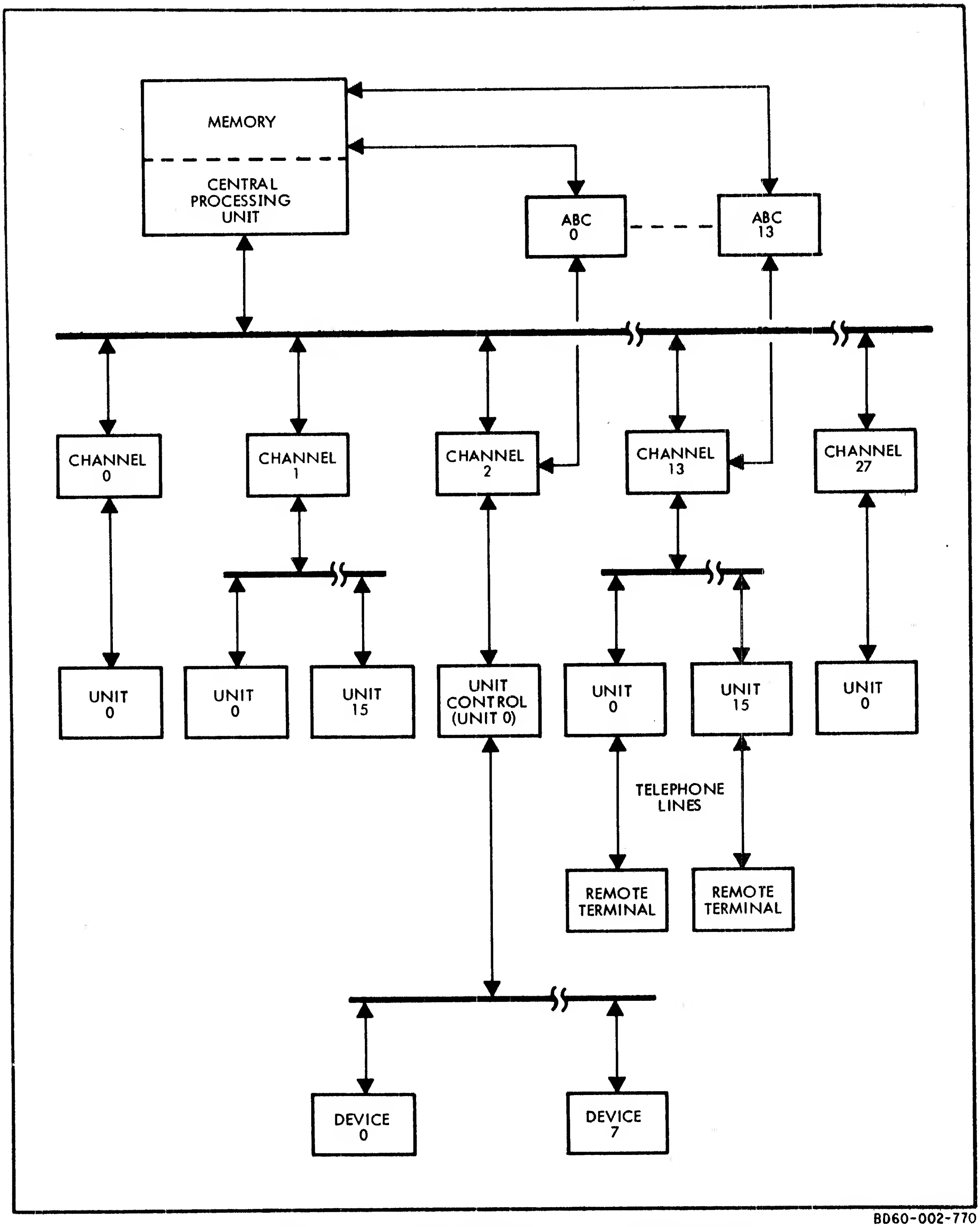


Figure 4-1. Series 6000 Computer I/O Structure

4-4 INPUT/OUTPUT COMMUNICATIONS

4-4.1 General

Input/output communications consist of the transfer of command, data, status or address words between the A register and the specified channel/unit combination upon the execution of appropriate input/output instructions. The channel/unit code in each I/O instruction (excluding Output Address Word (OAW) and Input Address Word (IAW) allows one of 28 channels to be selected and one of up to 16 units to be connected to that channel. When an instruction to the same channel carries a different unit code, the previously specified unit is disconnected and the new unit is connected automatically. During this disconnect/connect sequence, the channel is busy and does not respond to I/O instructions until the sequence is completed. If a channel is in the process of transferring commands or data to a unit, an Input Status Word (ISW) or Input Data Word (IDW) instruction addressed to a different unit on the same channel receives a busy indication (i.e., the Condition register is set to "Not Zero").

Command and data words from the CPU are transferred to the channel output buffer and subsequently to the connected peripheral unit. Data and status words are retrained in the input buffer of the selected peripheral unit and transferred to the A register upon request (instruction) from the CPU. Address words are applicable only to those channels employing an ABC (paragraph 4-4.5).

4-4.2 I/O Commands

The Output Command Word (OCW) instruction transfers a command word to the specified channel/unit combination. The command word bits specify the unit control function(s) to be performed and/or the I/O condition to be established. Following the execution of an OCW instruction, the channel remains busy until the command has been accepted by the addressed unit. The I/O command word format, shown in Figure 4-2, illustrates standard bit assignments.

If the channel is busy or not ready when addressed by the OCW instruction, the Condition register is set to "Not Zero" to allow a programmed delay. The Override function causes the channel to go through a unit disconnect/connect sequence regardless of the unit code. This clears the channel/unit of any other activity and allows the current instruction to assume control of the channel unconditionally upon termination of the disconnect/connect sequence.

A. Interrupt Control

Command word bits 0-2 are used to control the standard peripheral unit interrupts as defined in Table 4-1.

The two standard interrupts are defined as Input (Buffer Full) and Output (Buffer Empty). When these interrupts are enabled, the assigned interrupt level(s) will be triggered when the input buffer contains data for transfer to the CPU or when the output buffer is ready to receive a data or command word from the CPU.

Bits 0-2 may also be used for controlling special unit interrupts. Special peripheral unit interrupts are defined, where applicable, in Section V.

PERIPHERAL UNIT	23	22	21	20	19	18	17	12	11	10	9	8	7	6	5	4	3	2	1	0
TELETYPE														CLEAR INPUT BUFFER	DISABLE INPUT	ENABLE KEYBOARD	ENABLE READER	SELECT OUTPUT INTERRUPT	SELECT INPUT INTERRUPT	
HIGH-SPEED PAPER TAPE READER														CLEAR INPUT BUFFER	DISABLE READER		ENABLE READER		SELECT INPUT INTERRUPT	
HIGH-SPEED PAPER TAPE PUNCH														CLEAR OUTPUT BUFFER		DISABLE PUNCH (POWER OFF)	ENABLE PUNCH (POWER ON)	SELECT OUTPUT INTERRUPT		(1) ENABLE SELECTED INTERRUPT
CARD READER															OFFSET STACKER	EJECT CARD	FEED CARD	SELECT TROUBLE INTERRUPT	SELECT INPUT INTERRUPT	(0) DISABLE SELECTED INTERRUPT
LINE PRINTER																1 (WORD MODE) 0 (CHARACTER MODE)	RESET PRINT CYCLE	SELECT TROUBLE INTERRUPT	SELECT PRINT COMPLETE INTERRUPT	
DISC	1 INITIATE ABC INPUT 0 1 INITIATE ABC OUTPUT 1 0 SPECIAL ACTION CODE 1 0 SEEK 0		DRIVE NO. (0-7)			CYLINDER NUMBER (0-202)				TRACK (HEAD) NO. (0-19)					RECORD NO. (0-63)					

Figure 4-2. I/O Command Word Format

Table 4-1. I/O Interrupt Control

Bit Configuration 2	1	0	ACTION
0	0	0	No Action
0	0	1	No Action
0	1	0	Disable Input Interrupt
0	1	1	Enable Input Interrupt
1	0	0	Disable Output Interrupt
1	0	1	Enable Output Interrupt
1	1	0	Disable Both Interrupts
1	1	1	Enable Both Interrupts

B. Unit Commands

The specific unit commands, defined by bits 3-21, vary according to the requirements of the individual peripheral units. Refer to the peripheral unit descriptions in Section V for the definitions of the command bits.

C. ABC Control

The ABC control bits (22 and 23) in the command word are applicable only to those channels operating with an Automatic Block Controller. See paragraph 4-4.5 for the ABC description.

4-4.3 I/O Status

The Input Status Word (ISW) instruction is used to test the operational status of a peripheral unit. When a channel is addressed by the ISW instruction, a 24-bit status word is transferred to the A register in the CPU. The basic I/O status word format is illustrated on Figure 4-3.

The quantity and significance of the status bits depends on the type of peripheral unit involved. The WC bit (bit 23) is applicable only to those channels employing an ABC (paragraph 4-4.5).

4-4.4 Single Word Data Transfer

A. Input

The Input Data Word (IDW) instruction is a request from the CPU to a specific channel/unit combination for a data word. If data is available, the data word is transferred immediately to the A register. If data is not available, the Condition register is set to "Not Zero" to allow a programmed delay.

Normally, the 24-bit input data word contains a single data character. The actual number of data bits per character depends on the peripheral unit involved in the transfer. For example, the console typewriter generates an 8-bit character and the card reader generates a

PERIPHERAL UNIT	23	22	7	6	[illegible]	[illegible]	[illegible]	2	1	0
TELETYPE						KEYBOARD ENABLED	READER ENABLED	OUTPUT BUFFER EMPTY *	INPUT BUFFER FULL *	UNIT ON LINE
HIGH-SPEED PAPER TAPE READER					GATE OPEN		READER ENABLED		INPUT BUFFER FULL *	UNIT ON LINE
HIGH-SPEED PAPER TAPE PUNCH					TAPE LOW			OUTPUT BUFFER EMPTY *		UNIT ON (POWER ON)
CARD READER				COMMAND BUFFER FULL	NON PICK	STACKER FULL	HOPPER EMPTY	READER TROUBLE *	INPUT BUFFER FULL *	UNIT ON LINE
LINE PRINTER					STOP MODE ENABLED	BUSY	PARITY ERROR	PRINTER TROUBLE (PAPER LOW, YOKE OPEN)	PRINT COMPLETE *	UNIT ON LINE
DISC	ABC BUSY		BUSY (1) * READY (0)	INVALID RECORD SPECIFI-CATION	EOF	READ ERROR	SEEK ERROR	FILE UNSAFE	SELECTED DRIVE FILE PROTECTED	SELECTED DRIVE ON LINE

* THESE CONDITIONS TRIGGER APPROPRIATE INTERRUPT

Figure 4-3. I/O Status Word Format

12-bit character. In any case, the character is right-justified in the A register with the unused bit positions set to ZEROs.

Assuming the data character contains no more than 12 bits, more than one character may be packed in the A register through the use of the MERGE feature. When a character MERGE is employed, a logical OR is performed between the previous contents of the A register and the new input data word. Without the MERGE, the previous contents of A are destroyed upon transfer of a new character to A. An illustration of the character MERGE technique, as compared to a normal IDW instruction, is shown on Figure 4-4.

B. Output

When an Output Data Word (ODW) instruction is executed, a 24-bit data word is transferred from the A register to the specified channel. The data word is subsequently transferred from the channel to the unit that is currently connected. If the channel is busy or not ready to accept the data word, the Condition register is set to "Not Zero" to allow a programmed delay. If the unit is not ready to accept the data from the channel, the data remains in the channel buffer

As soon as the peripheral unit is able to accept the data from the channel, the channel-to-unit transfer is made, thereby freeing the channel buffer for another data (or command) word from the CPU.

The number of data bits accepted by the peripheral unit varies according to the type of unit involved. Some peripheral devices are word-oriented and accept the entire 24-bit word. Others are character-oriented and accept only a specific number of bits per character.

EXAMPLE: THREE 8-BIT DATA CHARACTERS ARE TO BE PACKED IN THE A REGISTER.

(a) NORMAL (WITHOUT MERGE)

CODING	COMMENTS	REGISTER A 23 ... 8	REGISTER A 7 ... 0
IDW CU	BRING IN FIRST DATA CHARACTER	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	C_1
. . .			
IDW CU	BRING IN SECOND CHARACTER	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	C_2
. . .			
IDW CU	BRING IN THIRD CHARACTER	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	C_3
. . .			

(b) MERGE

CODING	COMMENTS	REGISTER A 23 ... 16	REGISTER A 15 ... 8	REGISTER A 7 ... 0
IDW CU	BRING IN FIRST DATA CHARACTER	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	C_1
LLA 8	SHIFT LEFT 8 PLACES	0 0 0 0 0 0 0 0	C_1	0 0 0 0 0 0 0 0
IDW* CU	BRING IN SECOND CHARACTER AND MERGE	0 0 0 0 0 0 0 0	C_1	C_2
LLA 8	SHIFT LEFT 8 PLACES	C_1	C_2	0 0 0 0 0 0 0 0
IDW* CU	BRING IN THIRD CHARACTER AND MERGE	C_1	C_2	C_3
. . .				

MI60-005-770

Figure 4-4. IDW Instruction, Data Character Formatting

4-4.5 Automatic Block Transfer

A block data transfer is controlled by the Automatic Block Controller (ABC). A block transfer is initiated under program control and proceeds from that point without further program intervention until the transfer is completed or until another transfer is initiated.

Two ABC registers initialize and subsequently control a block transfer. They are the Transfer Address Register (TAR) and the Word Count Register (WCR). TAR contains the address of the first word in a block of data. The WCR defines the number of words to be transferred.

An Automatic Restart Flag (ARF) permits the sequential transfer of more than one block of data. If this flag is set when the initial data block transfer is completed, the next data block will be transferred without program intervention.

The ABC is initialized by loading the address of the first of a pair of block transfer parameters in the TAR with an ODW instruction. These parameters are stored in sequential memory locations and are illustrated on Figure 4-5. The OAW instruction transfers the contents of register A to the TAR. If an ABC transfer is in process, it will be terminated. An OAW instruction does not clear the I/O buffers nor does it initiate a block transfer.

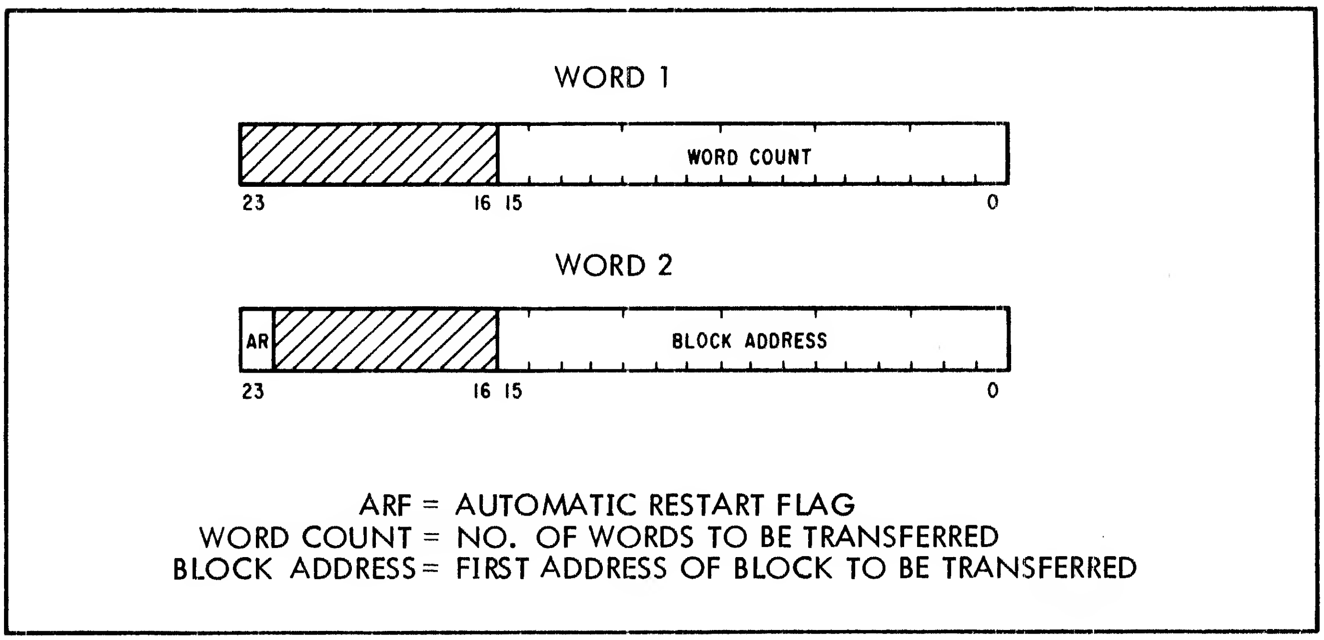


Figure 4-5. Block Transfer Parameters, Storage Format

An automatic block transfer is initiated by executing an OCW instruction with bits 22 and 23 of the command word set as shown on Figure 4-6. On detecting a command to transfer data, the ABC loads the WCR, TAR, and the ARF from the previously specified parameters.

Once a block transfer has been initiated, n words (n = Word Count) are transferred automatically. After each word is transferred, the contents of the TAR are incremented by one and the contents of the WCR are decremented by one. When the block transfer is complete

Bit 23	Bit 22	Action
1	0	Initiate Input Transfer
1	1	Initiate Output Transfer

Figure 4-6. Block Transfer Directional Control Format

(Word Count = 0), the TAR is loaded with the contents of n+1. If the ARF is currently set, the address obtained from n+1 is assumed to be a new parameter address and a new transfer sequence is initiated without program intervention. If the ARF is not set, the transfer sequence is discontinued and the contents of n+1 remains in the TAR. An interrupt is generated after the TAR is loaded with n+1, regardless of the state of the ARF. Refer to Figure 4-7 for the ABC logic.

The ABC channel answers busy until the block transfer is terminated, except for an ISW or IAW instruction. When in the output mode, the channel may answer busy after the block transfer is completed between the CPU and the ABC channel. This is because the peripheral unit has not accepted the data. The ABC remains busy until the data is taken by the peripheral unit.

An ISW instruction may be used to determine whether the Word Count is complete or not. Bit 23 of the status word, when set (ONE) indicates that the Word Count is not complete. When bit 23 of the status word is ZERO, the Word Count = 0.

The IAW instruction is used to transfer the current contents of TAR to register A. This instruction may be executed at any time without disturbing the block transfer sequence.

The OCW instruction, which initiates the block transfer, can also be used to terminate the transfer sequence by setting the Override bit (bit 11). This halts the transfer sequence with the contents of TAR and WCR intact (at the point of termination).

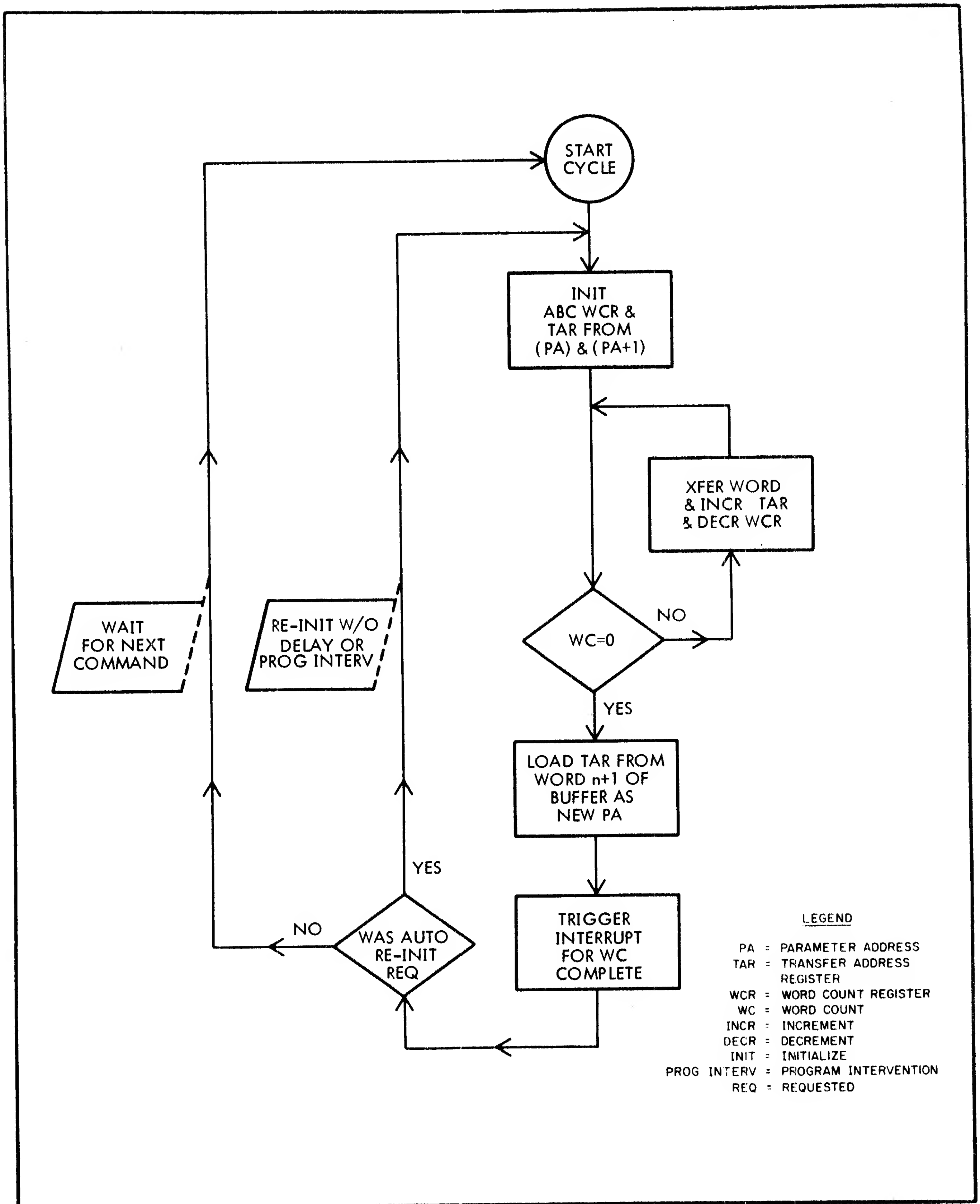
The following examples illustrate three different ABC applications.

Example 1: Simple, single buffer input

```
        TOA    PA      Parameter Address
        OAW    C       Initialize TAR
        TMA    CW      Command Word
        OCW    CU      Initiate transfer
        BNZ    *-1     Delay if channel is busy
        ...
CW      DATA           Bit 23 and others as required by the I/O device
PA      DAC    n       Absolute Word Count
        DAC    BUFF    Address of input buffer
BUFF    BLOK   n       Reserve n words. Word n+1 is of no significance
                       since the AR bit is not set.
```

Example 2: Multi-buffered output with automatic restart and buffer switching.

```
        TOA    PA1     Parameter Address 1
        OAW    C       Initialize TAR
        TMA    CW      Command Word
        OCW    CU      Initiate first transfer
```



MI60-036-770

Figure 4-7. ABC Logic

```
         BNZ     *-1      Delay if channel is busy
         ...
CW       DATA             Bits 23, 22, and others as required by the I/O device
PA1      DAC     n        Word Count
         DAC*    BUF1     Address of buffer 1 and the ARF (*)
PA2      DAC     n        Word Count
         DAC*    BUF2     Address of buffer 2 and the ARF (*)
BUF1     BLOK    n        Reserve n words
         DAC     PA2      Automatic Reinitialization address for TAR, to
                          switch buffers
BUF2     BLOK    n        Reserve n words
         DAC     PA1      Automatic Reinitialization address for TAR, to
                          switch buffers
```

NOTE

Once this cycle is initiated it will continue, without program intervention, until a new command is received.

Example 3: Multi-buffer input with automatic buffer switching but without Automatic Restart.

```
         TOA     PA1      Parameter Address 1
         OAW     C        Initialize TAR
         TOI     BUF1     Setup pointer for buffer to be processed
         TIM     BA
         TOI     BUF2     Prepare for initial interchange
         TMA     CW       Command Word
         OCW     CU       Initiate first transfer
         BNZ     *-1      Wait until channel takes command
SBAR     IMI     BA       Switch process pointer
         TMA     CW       Command Word
         OCW     CU       Initiate next transfer
         BNZ     *-1      As soon as channel is free
         ...              Process current buffer, whose base address is in
         ....             register I, while the alternate buffer is being
         ...              filled
         ...
         BUC     SBAR     To switch buffers and restart ABC
         ...
CW       DATA             Bits 15, 14 and others required by I/O device
BA       ZZZ              Alternating buffer address
PA1      DAC     n        Word count only
         DAC     BUF1     Buffer 1 address
PA2      DAC     n        Word Count
         DAC     BUF2     Buffer 2 address
BUF1     BLOK    n        Reserve n words
         DAC     PA2      Automatic Reinitialization address for TAR
BUF2     BLOK    n        Reserve n words
         DAC     PA1      Automatic Reinitialization address for TAR
```

NOTE

This example illustrates an application in which the processing rate is slower than the input rate via ABC.

SECTION V

PERIPHERAL UNITS

5-1 INTRODUCTION

All Series 6000 peripheral units are designed for compatibility with the computer input/output structure to permit maximum program control of unit operations. Communications between the computers and the standard peripheral units are controlled by the standard set of I/O instructions. (Reference: Section IV)

The following pages provide a description of the standard Series 6000 peripheral units. The major paragraphs in this section describe the types of peripheral units available and the features of each unit.

5-2 KEYBOARD/PRINTER MODELS 6001-1 AND 6001-2

5-2.1 General

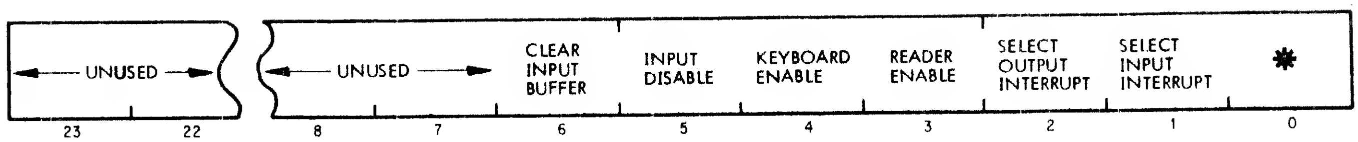
The Model 6001-1 Keyboard/Printer consists of a Teletype Corporation ASR-33 (Automatic Send-Receive) teletypewriter set equipped with an integral 8-level paper tape punch and reader. Model 6001-2 is a Teletype Corporation ASR-35 teletypewriter equipped with similar equipment. They provide input data to the computer from the typewriter keyboard or paper tape reader. Computer output data is processed by the typing unit or paper tape punch. The ASR-33 and ASR-35 are equipped with a Datacraft controller to provide communications with the computer under program control.

5-2.2 Computer Communications

The ASR-33 and ASR-35 can be used in both on-line and off-line operations. In the on-line mode, the unit operates in conjunction with the computer under program control. The off-line mode allows the paper tape reader and punch to duplicate an existing tape or the keyboard and paper tape punch may be employed to prepare a tape. A facility for monitoring the tape preparation is provided by the typing unit.

5-2.3 Command Word

On-line operation of the teletypewriter is initiated by the transfer of a command word from the computer to the teletypewriter. The command word performs various setup and control functions. The command word format is illustrated on Figure 5-1. Definitions of each command word bit are also provided in Table 5-1.



*Bit 0 = 0 - Disable Selected Interrupt
Bit 0 = 1 - Enable Selected Interrupt

Figure 5-1. ASR-33 and ASR-35 Teletypewriter Command Word Format

Table 5-1. ASR-33 and ASR-35 Teletypewriter Command Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Interrupt Control	Enables/disables selected interrupt (bit 1 or bit 2)
1	Select Input Interrupt	Allows the unit to generate an input interrupt when enabled by bit 0 (interrupt control). The input interrupt is activated when the input data buffer contains data ready for transfer to the computer.
2	Select Output Interrupt	Allows the unit to generate an output interrupt when enabled by bit 0 (interrupt control). The output interrupt is activated when the output data buffer is empty and ready to receive a data or command word from the computer. NOTE Each teletypewriter interrupt operates as an independent function. They may be selected separately or together; however, they must be enabled or disabled simultaneously.
3	Reader Enable	Initiates a sequence that advances the paper tape for reading and loads the detected character into the input buffer. Once enabled, the paper tape reader operates automatically as long as each character is accepted by the computer or until the reader is disabled by a new command.
4	Keyboard Enable	Allows a character, resulting from the depression of a typewriter key, to be loaded into the input buffer. NOTE The Keyboard Enable command locks out the paper tape reader; however, the keyboard is not disabled when the Reader Enable command is executed. The typewriter keys are capable of functioning while the paper tape reader is operating. Since the paper tape reader and keyboard share the same data lines, caution should be exercised so that a typewriter key is not activated. This could load an erroneous character into the input data buffer.

Table 5-1. ASR-33 and ASR-35 Teletypewriter Command Word Bit Description (Cont'd.)

BIT	FUNCTION	DESCRIPTION
5	Input Disable	Disables the presently enabled input device (paper tape reader or keyboard) and prohibits further input operations until another Keyboard Enable or Reader Enable command is received. NOTE The Reader Enable, Keyboard Enable and Input Disable commands (Bits 3, 4, and 5) do not clear the input data buffer.
6	Clear Input Buffer	Clears the input data buffer, but does not disturb the enabled state of the paper tape reader or keyboard.
7-23	Spare	Reserved for future expansion.

5-2.4 Status Word

The computer can determine the operational status of the unit at any time by transferring a status request to the unit. A status word is formed and returned to the computer. With the exception of the mutually exclusive Reader Enabled and Keyboard Enabled bits, the status bits are independent and may exist simultaneously. Figure 5-2 illustrates the status word format for the teletypewriter. Definitions of the status word bits are also provided in Table 5-2.

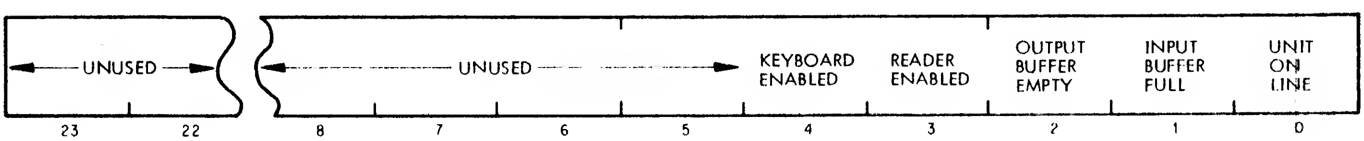


Figure 5-2. ASR-33 and ASR-35 Teletypewriter Status Word Format

Table 5-2. ASR-33 and ASR-35 Teletypewriter Status Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Unit On-Line	Signifies the unit is connected to the input/output channel and power is on.
1	Input Buffer Full	Indicates the input data buffer contains a word ready for transfer to the computer. This condition triggers the Input Interrupt when that interrupt is enabled.

Table 5-2. ASR-33 and ASR-35 Teletypewriter Status Word Bit Description (Cont'd.)

BIT	FUNCTION	DESCRIPTION
2	Output Buffer Empty	Indicates the output data buffer is empty and ready to receive a command or data word from the computer. This condition triggers the Output Interrupt when that interrupt is enabled.
3	Reader Enabled	Indicates the paper tape reader is enabled and will read the tape and load the resulting character in the input data buffer.
4	Keyboard Enabled	Indicates the typewriter keyboard is enabled and loads a character into the input data buffer when a key is depressed.
5-23	Spare	Reserved for future expansion.

5-2.5 Data Transfer

The Teletypewriter is character oriented and operates with characters containing up to eight bits. A character generated by the paper tape reader or keyboard is loaded into the input data buffer prior to transfer to the computer. A character generated by the computer is loaded into the output data buffer and subsequently operates the typewriter's typing unit and paper tape punch. Both input and output characters are right justified in the 24-bit data word.

For input operations, the paper tape reader and keyboard function as separate and mutually exclusive devices. When executing output data transfers, the computer regards the typing unit and paper tape punch as a single device. The typing unit and paper tape punch function simultaneously, unless the punch is manually turned off. All data characters will be punched regardless of whether it is a printable character or a control character. All printing, punching and carriage control functions are initiated by a data character.

5-2.6 Characteristics

The major operating characteristics of the ASR-33 and ASR-35 Teletypewriters are listed below.

Typing Unit

Number of printable characters	63
Number of characters per line	72
Operating Speed	10 characters/second
Vertical Spacing	6 lines/inch
Horizontal Spacing	10 characters/inch
Page Required	Standard roll, 8 1/2 inches wide, five inches in diameter

Paper Tape Punch

Levels	8
Operating Speed	10 characters/second
Bits per character	8 bits (ASCII code)
Tape packing density	10 characters/inch

Paper Tape Reader

Operating Speed	10 characters/second (OFF-LINE operation) 20 characters/second (ASR-33) 10 characters/second (ASR-35) (ON-LINE operation)
Tape	1-inch wide paper or mylar
Tape Leader	3-inch minimum

5-2.7 Character Set

The ASR-33 and ASR-35 character set is defined in Table 5-3.

Table 5-3. ASR-33 and ASR-35 Character Set

Symbol or Function	Octal Code	Symbol or Function	Octal Code	Symbol or Function	Octal Code
A	301	Y	331	$	244
B	302	Z	332	%	245
C	303			&	246
D	304	0	260	'	247
E	305	1	261	(	250
F	306	2	262	)	251
G	307	3	263	*	252
H	310	4	264	+	253
I	311	5	265	,	254
J	312	6	266	-	255
K	313	7	267	.	256
L	314	8	270	/	257
M	315	9	271	:	272
N	316			;	273
O	317	@	300	<	274
P	320	[	333	=	275
Q	321	\	334	>	276
R	322	]	335	?	277
S	323	↑	336	Carriage Return*	215
T	324	←	337	Line Feed*	212
U	325	SPACE	240	Bell*	207
V	326	!	241	Delete*	377
W	327	"	242		
X	330	#	243	*Does not produce a printed character.	

5-3 KEYBOARD/PRINTER MODEL 6001-3

Model 6001-3 Keyboard/Printer consists of a Teletype Corporation KSR-35 (Keyboard Send-Receive) teletypewriter set and a Datacraft Controller. Model 6001-3 operates in a manner similar to Models 6001-1 and 6001-2 (refer to paragraph 5-2) except it is not equipped with a paper tape punch or reader. The KSR-35 teletypewriter operates only in the on-line mode.

The command and status word formats and bit descriptions are the same as those for the Models 6001-1 and 6001-2. Refer to Figure 5-1 and 5-2 for the command and status word formats respectively. Tables 5-1 and 5-2 describe the functions of the command and status word bits.

5-4 PAPER TAPE READER MODELS 6002-10, 6002-20 AND 6002-21 (SPOOLER)

5-4.1 General

The paper tape reader is a Digitronics Model 2540 Series Perforated Tape Reader. It reads from five to eight-level perforated tape at speeds of 300 (Model 6002-10) or 600 (Model 6002-20) characters per second. The paper tape reader operates in conjunction with Datacraft designed interface logic (controller) to allow the unit to operate under computer control. An optional spooler (Model 6002-21) provides tape handling.

5-4.2 Computer Communications

The paper tape reader operates in conjunction with the DC 6024 Computer as an input peripheral device. The reader operates under program control by using command and status words.

5-4.3 Command Word

Operation of the paper tape reader is initiated by transferring a command word from the computer to the reader. The command word performs various setup and control functions and is shown on Figure 5-3. A brief description of each command word bit is also provided in Table 5-4.

← UNUSED →	← UNUSED →	CLEAR INPUT BUFFER	READER DISABLE	UNUSED	READER ENABLE	UNUSED	SELECT INPUT INTERRUPT	*
23 — 22	8 — 7	6	5	4	3	2	1	0

*Bit 0 = 0 - Disable Input Interrupt
Bit 0 = 1 - Enable Input Interrupt

Figure 5-3. Paper Tape Reader Command Word Format

Table 5-4. Paper Tape Reader Command Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Interrupt Control	Enables/disables input interrupt (bit 1).
1	Select Input Interrupt	Allows the unit to generate an input interrupt when the input data buffer is loaded and ready to transfer data to the computer. The input interrupt must be enabled or disabled by bit 0 (interrupt control).
2	Spare	Reserved for future expansion.
3	Reader Enable	Initiates a sequence that advances the paper tape for reading and loads the detected character in the input data buffer.
4	Spare	Reserved for future expansion.
5	Reader Disable	Disables the input capability of the paper tape reader and prohibits further input operations until another Reader Enable command is received. NOTE: The Reader Enable and Reader Disable commands (bits 3 and 5) do not clear the input data buffer.
6	Clear Input Buffer	Clears the input data buffer but does not disturb the enabled state of the paper tape reader.
7-23	Spare	Reserved for future expansion.

5-4.4 Status Word

The computer can determine the operational status of the paper tape reader unit at any time by transmitting a status request to the unit. A status word is formed and transmitted to the computer. All status bits are independent and may exist simultaneously. Definitions for each status word bit are provided in Table 5-5 along with the status word format (Figure 5-4).

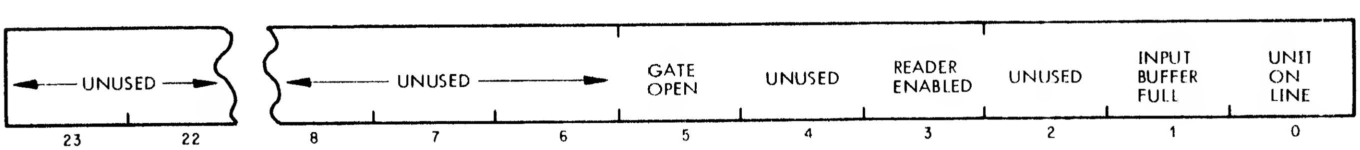


Figure 5-4. Paper Tape Reader Status Word Format

Table 5-5. Paper Tape Reader Status Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Unit On-Line	Signifies the unit is connected to the input/output channel and power is on.
1	Input Buffer Full	Indicates the input data buffer contains a word ready for transfer to the computer. This condition triggers the Input Interrupt when that interrupt is enabled.
2	Spare	Reserved for future expansion.
3	Reader Enabled	Indicates the paper tape reader is enabled and will read the tape and load the resulting characters in the input data buffer.
4	Spare	Reserved for future expansion.
5	Gate Open	Indicates the TAPE LOAD LEVER is in the LOAD position which disengages the front tape guides.
6-23	Spare	Reserved for future expansion.

5-4.5 Data Transfer

The paper tape reader operates with 8-bit characters. A character, generated by reading the perforated tape, is read into the input data buffer. The 8-bit character is subsequently transferred to the computer via the input/output channel. Each character is right justified in the 24-bit data word.

5-4.6 Characteristics

The Model 6002 Paper Tape Reader has the major operating characteristics listed below.

Drive	Unidirectional
Operating Speed	300 characters/second (Model 6002-10) 600 characters/second (Model 6002-20)
Tape	Paper, Paper-Mylar, aluminized mylar or solid mylar; .0025 to .005 inch thick.

5-4.7 Character Set

The character set for the paper tapereader is provided in Table 5-6.

Table 5-6. Model 6002 Paper Tape Reader Character Set

Symbol or Function	Octal Code	Symbol or Function	Octal Code	Symbol or Function	Octal Code
A	301	Y	331	$	244
B	302	Z	332	%	245
C	303			&	246
D	304	0	260	'	247
E	305	1	261	(	250
F	306	2	262	)	251
G	307	3	263	*	252
H	310	4	264	+	253
I	311	5	265	,	254
J	312	6	266	-	255
K	313	7	267	.	256
L	314	8	270	/	257
M	315	9	271	:	272
N	316			;	273
O	317	@	300	<	274
P	320	[	333	=	275
Q	321	\	334	>	276
R	322	]	335	?	277
S	323	↑	336	Carriage Return*	215
T	324	←	337	Line Feed*	212
U	325	SPACE	240	Bell*	207
V	326	!	241	Delete*	377
W	327	"	242		
X	330	#	243		
				*Does not produce a printed character.	

5-5 PAPER TAPE PUNCH MODEL 6003-1

5-5.1 General

Datacraft Model 6003-1 Paper Tape Punch consists of a Teletype Corporation High-Speed BRPE Paper Tape Punch and a Datacraft controller. The punch operates at speeds up to 110 characters per second to record coded data on perforated tape. The controller provides an interface between the computer and the punch that allows the punch to be controlled by the computer.

5-5.2 Computer Communications

The paper tape punch operates in conjunction with the DC 6024 computer as an output peripheral device. Communications between the computer and the punch are controlled by command and status words generated by the computer.

5-5.3 Command Word

A command word from the computer initiates operation of the paper tape punch. The command word format is illustrated on Figure 5-5. It performs various setup and control functions in the peripheral unit. Also provided is a brief description of the function performed by each bit of the command word in Table 5-7.

23 – 22	8 – 7	6	5	4	3	2	1	0
← UNUSED →	← UNUSED →	CLEAR OUTPUT BUFFER	UNUSED	PUNCH DISABLE (POWER OFF)	PUNCH ENABLE (POWER ON)	SELECT OUTPUT INTERRUPT	UNUSED	*

*Bit 0 = 0 - Disable Output Interrupt
Bit 0 = 1 - Enable Output Interrupt

Figure 5-5. Paper Tape Punch Command Word Format

Table 5-7. Paper Tape Punch Command Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Interrupt Control	Enables/disables the output interrupt (bit 2).
1	Spare	Reserved for future expansion.
2	Select Output Interrupt	Allows the unit to generate an output interrupt when the output data buffer is empty and capable of receiving data from the computer. The output interrupt must be enabled or disabled by bit 0 (interrupt control).
3	Punch Enable (Power On)	Applies power to the punch unit and prepares the unit to receive and process data.
4	Punch Disable (Power Off)	Removes power from the punch unit and returns it to a standby status.
5	Spare	Reserved for future expansion.
6	Clear Output Buffer	Clears the output data buffer.
7-23	Spare	Reserved for future expansion.

5-5.4 Status Word

The computer can determine the operational status of the paper tape punch unit at any time by transmitting a status request to the unit. A status word is formulated and returned to the computer indicating the condition of the unit. Each status bit is independent and may exist simultaneously. Figure 5-6 illustrates the status word format. Also provided are definitions of the status bits in Table 5-8.

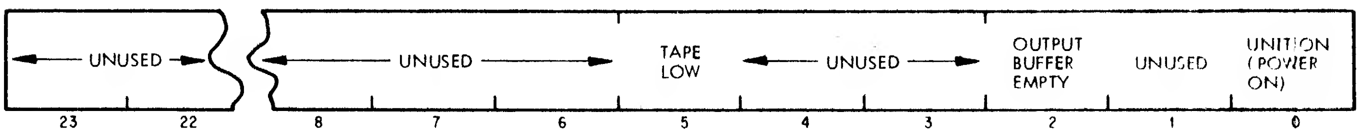


Figure 5-6. Paper Tape Punch Status Word Format

Table 5-8. Paper Tape Punch Status Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Unit On (Power On)	Indicates the paper tape punch unit has power applied.
1	Spare	Reserved for future expansion.
2	Output Buffer Empty	Indicates the output data buffer is empty and ready to process a command or data word from the computer. This condition triggers an Output Interrupt when that interrupt is enabled.
3-4	Spare	Reserved for future expansion.
5	Tape Low	Indicates the supply of paper tape is about to run out.
6-23	Spare	Reserved for future expansion.

5-5.5 Data Transfer

The paper tape punch accepts 8-bit characters from the computer and stores them in the output data buffer. The characters are decoded and converted into holes (or non-holes) on a paper tape.

5-5.6 Characteristics

The major operating characteristics of the paper tape punch are as follows:

Punching Speed	110 characters/second
Tape Levels	5, 6, 7, or 8 with in-line feed holes and 6 with advanced feed hole.
Tape Widths	5 level (11/16 inch), 6 and 7 level (7/8 inch), and 8 level (1 inch)

5-5.7 Character Set

The character set applicable to the paper tape punch is provided in Table 5-9.

Table 5-9. Model 6003-1 Paper Tape Punch Character Set

Symbol or Function	Octal Code	Symbol or Function	Octal Code	Symbol or Function	Octal Code
A	301	Y	331	$	244
B	302	Z	332	%	245
C	303			&	246
D	304	0	260	'	247
E	305	1	261	(	250
F	306	2	262	)	251
G	307	3	263	*	252
H	310	4	264	+	253
I	311	5	265	,	254
J	312	6	266	-	255
K	313	7	267	.	256
L	314	8	270	/	257
M	315	9	271	:	272
N	316			;	273
O	317	@	300	<	274
P	320	[	333	=	275
Q	321	\	334	>	276
R	322	]	335	?	277
S	323	↑	336	Carriage Return*	215
T	324	←	337	Line Feed*	212
U	325	SPACE	240	Bell*	207
V	326	!	241	Delete*	377
W	327	"	242		
X	330	#	243		
				*Does not produce a printed character.	

5-6 LINE PRINTER MODEL 6004-3

5-6.1 General

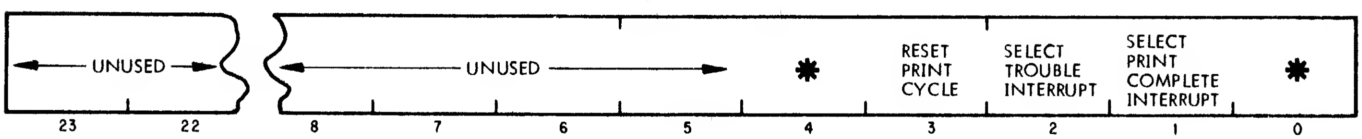
Datacraft's Model 6004-3 Line Printer operates under program control to provide a permanent record of output data from the computer. Printing functions are performed by an electro-mechanical print mechanism and paper feed system. The printer is equipped with a magnetic core memory capable of storing 137 characters. Each character contains seven bits, six data bits and a parity bit. The Model 6004-3 Line Printer consists of a Mohawk Data Sciences Corporation 5000 Series Printer and a Datacraft built interface controller.

5-6.2 Computer Communications

Communications between the computer and the line printer are controlled by command and status words. Command words initiate and control the printing operation. Status words inform the computer of the line printer's operating status.

5-6.3 Command Word

Transfers of a command word to the line printer controls the interrupt functions and sets up the line printer for an operation. The command word format is illustrated on Figure 5-7. Each bit and its function are also described in Table 5-10.



*Bit 0 = 0 - Disable Selected Interrupt
Bit 0 = 1 - Enable Selected Interrupt

*Bit 4 = 0 - Character Mode
Bit 4 = 1 - Word Mode

Figure 5-7. Line Printer Command Word Format

Table 5-10. Line Printer Command Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Interrupt Control	Enables or disables the interrupt selected by bit 1 (print complete interrupt) or bit 2 (trouble interrupt). When bit 0 is a logic 1, the selected interrupt is enabled. Conversely when bit 0 is a logic 0, the selected interrupt is disabled.

Table 5-10. Line Printer Command Word Bit Description (Cont'd.)

BIT	FUNCTION	DESCRIPTION
1	Select Print Complete Interrupt	Allows the Print Complete Interrupt to be enabled or disabled by bit 0 (interrupt control). If selected and enabled, Print Complete generates a trigger on the assigned interrupt level when the line printer print cycle is completed.
2	Select Trouble Interrupt	Allows the line printer Trouble Interrupt to be enabled or disabled by bit 0 (interrupt control). If selected and enabled, a trigger is generated on the assigned interrupt level when a malfunction occurs in the line printer. Examples of line printer troubles are a low paper level, no paper condition, or an open line printer yoke. NOTE Each line printer interrupt operates as an independent function. They may be selected separately or together; however, they must be enabled or disabled simultaneously.
3	Reset Print Cycle	Establishes the necessary initial condition for the line printer when beginning an operation or when the printer is to be setup for a carriage control function.
4	Mode Select	Determines if the line printer will be character oriented or word oriented. If the word mode is selected, each word received by the line printer is disassembled to form three characters.
5-23	Spare	Reserved for future expansion.

5-6.4 Status Word

Operational status of the line printer is available to the computer at all times. To obtain this status, the computer transfers a status request to the line printer unit. Each bit in the status word is independent and may exist simultaneously. Figure 5-8 illustrates the status word format and Table 5-11 provides a brief description of each status word bit.

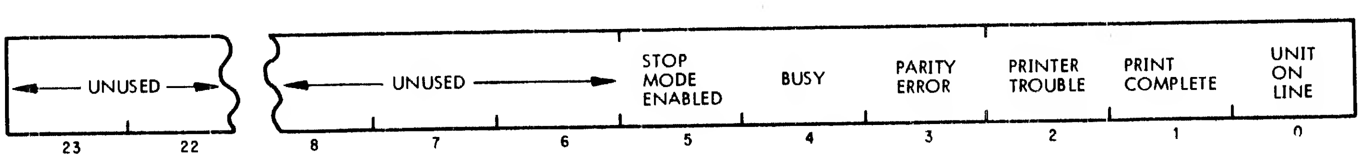


Figure 5-8. Line Printer Status Word Format

Table 5-11. Line Printer Status Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Unit On Line	Signifies the line printer unit is connected to the channel and that unit power is on.
1	Print Complete	Indicates the line printer has completed its print cycle.
2	Printer Trouble	Indicates a problem exists within the line printer. This indication may result from a low paper supply, a no-paper condition (paper break), the line printer yoke being open, or any other alarm condition.
3	Parity Error	Indicates the detection of a parity error within the line printer. Detection of a parity error inhibits printing of the affected characters.
4	Busy	Indicates the printer is in the process of a print cycle.
5	Stop Mode Enabled	Indicates the line printer is in a stop mode and all operations have been halted.
6-23	Spare	Reserved for future expansion.

5-6.5 Data Transfer

The line printer accommodates a maximum of 138 characters (C_0-C_{137}) per line. Two of the 138 characters are not printed but are used for control purposes. Character C_0 is always a carriage control character which specifies either the number of lines to be advanced prior to printing (C_0 = logic 0) or the channel number of the printer's vertical format control tape (C_0 = logic 1).

Table 5-12 provides examples of several line printer characters and describes the reactions caused in the line printer when they are used for carriage control.

The code 36_8 is reserved for use as the PRINT character. When this character is received by the line printer, characters C_1-C_n($n \leq 136$) are printed. The PRINT character also causes the print cycle to be set to C_0 prior to loading new data into the line printer memory. A Reset Print Cycle command also accomplishes this function.

Data characters (C_1, C_2 . . . . C_n) are loaded into sequential locations in the line printer memory. If an entire line is to be printed, data characters are loaded into the first 136 locations and the PRINT character, if present, is loaded into location 137. If n characters (< 136) are to be printed, the first n locations are used for the data and location n+1 is used for the PRINT characterif present. If an entire line (C_1-C_{136}) is received without the PRINT character, printing is automatically initiated and C_0 is set to 21_8 for channel 1 control.

Table 5-12. Carriage Control Character Functions

Character	Octal Code	Effect
@	00	0 line advance
A	01	1 line advance
B	02	2 line advance
.	.	.
.	.	.
.	.	.
.	.	.
O	17	15 line advance
P	20	Channel 0 control*
Q	21	Channel 1 control**
R	22	Channel 2 control
S	23	Channel 3 control
T	24	Channel 4 control
U	25	Channel 5 control
V	26	Channel 6 control
W	27	Channel 7 control

*Top of form channel
**Continuous, unformatted print channel

5-6.6 Characteristics

Operating characteristics of the Model 6004-3 Line Printer are listed below.

Printing Speed	1000 lines/minute
Characters	62 (plus BLANK and PRINT)
Bits per character	6 (truncated ASCII code)
Characters per line	136 (maximum) program-controlled
Form (paper) width	4-20 inches

5-6.7 Character Set

The line printer character set is listed in Table 5-13.

5-7 CARD READER MODELS 6005-1 AND 6005-2

5-7.1 General

The card reader is an input source for the computer. Input data is stored on the cards as a series of punched holes. The card reader consists of a supply hopper, photo-electric reading mechanism, and a stacker hopper. After the cards are loaded into the supply hopper, communications with the computer is by program control.

The Model 6005-1 Card Reader consists of a Mohawk Data Sciences Corporation Super Compact Card Reader (SCCR) Model 6002. The Model 6005-2 Card Reader consists of a Mohawk Data Sciences Corporation Card Reader Model 6021. Both models are equipped with a Datacraft controller.

Table 5-13. Line Printer Character Set

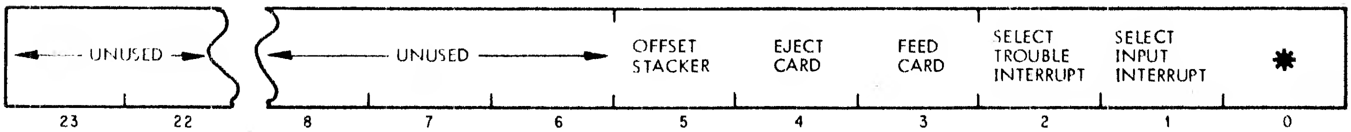
Symbol	Octal Code	Symbol	Octal Code
A	01	@	00
B	02	|	33
C	03	¤	34
D	04	¬	35
E	05	Print	36
F	06	——	37
G	07	Space	40
H	10	!	41
I	11	"	42
J	12	#	43
K	13	$	44
L	14	%	45
M	15	&	46
N	16	'	47
O	17	(	50
P	20	)	51
Q	21	*	52
R	22	+	53
S	23	,	54
T	24	-	55
U	25	.	56
V	26	/	57
W	27	:	72
X	30	;	73
Y	31	<	74
Z	32	=	75
		>	76
0	60	?	77
1	61		
2	62		
3	63		
4	64		
5	65		
6	66		
7	67		
8	70		
9	71		

5-7.2 Computer Communications

The card reader operates in conjunction with the DC 6024 Computer as an input peripheral device. Communications between the computer and the card reader are a result of program instructions that generate and monitor command and status words.

5-7.3 Command Word

Operation of the card reader is initiated by transferring a command word from the computer to the reader. The command word performs various setup and control functions and the format is illustrated on Figure 5-9. A brief description of the command word bits is provided in Table 5-14.



*Bit 0 = 0 - Disable Selected Interrupt
Bit 0 = 1 - Enable Selected Interrupt

Figure 5-9. Card Reader Command Word Format

Table 5-14. Card Reader Command Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Interrupt Control	Enables or disables the interrupt selected by bit 1 (input interrupt) or bit 2 (trouble interrupt). When bit 0 is a logic 1, the selected interrupt is enabled. Conversely when bit 0 is a logic 0, the selected interrupt is disabled.
1	Select Input Interrupt	Allows the input interrupt to be enabled or disabled by bit 0 (interrupt control). If selected and enabled, a trigger is generated on the assigned interrupt level when a data word is loaded in the input data buffer and ready for transfer to the computer.
2	Select Trouble Interrupt	Allows the trouble interrupt to be enabled or disabled by bit 0 (interrupt control). If selected and enabled, a trigger is generated on the assigned interrupt level when a malfunction occurs in the card reader. Examples of malfunctions are a pick failure, card motion error, a light current or dark current error, hopper empty, or stacker full. NOTE Each card reader interrupt operates as an independent function. They may be selected separately or together; however, they must be enabled or disabled simultaneously.

Table 5-14. Card Reader Command Word Bit Description (Cont'd.)

BIT	FUNCTION	DESCRIPTION
3	Feed Card	Transfers a single card from the card hopper to the reading mechanism. A feed card command must be generated for each card in the deck before it can be read. NOTE If a full 80-column is to be read, the feed card command is followed by 80 consecutive data transfer operations. If a partial card (less than 80 columns) is to be read, the eject card command can be used (refer to the description for bit 4).
4	Eject Card	Controls the number of card columns read by inhibiting the loading of data into the input data buffer. For example, if only the first 40 columns were to be read, the sequence would be as follows: (1) Feed card command (2) 40 data transfer operations (3) Eject card command NOTE In order for the eject card command to be effective, it must be executed within 400 microseconds (Model 6005-2) after the last desired column is read. This prevents the input data buffer from being loaded again.
5	Offset Stacker	Causes a card entering the stacker to be offset on the deck with a long edge protruding approximately 1/4 inch. This provides for ready, visual identification of the card by the operator. NOTE The offset stacker command may be executed at any time during the card's pass through the reading mechanism. It must occur not later than one millisecond after column 80 is read.
6-23	Spare	Reserved for future expansion.

5-7.4 Status Word

The computer can determine the status of the unit at any time by transferring a status request to the unit causing a status word to be transferred to the computer. Each bit in the status word is independent and may exist simultaneously. Figure 5-10 illustrates the status word format and Table 5-15 provides a brief description of the status word bits.

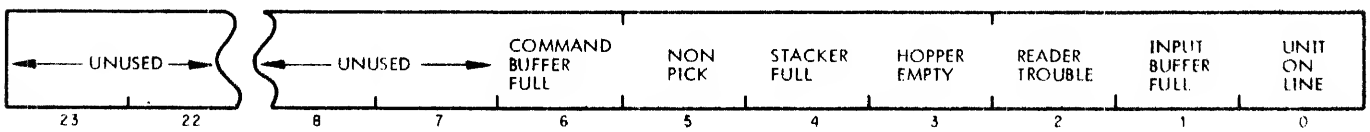


Figure 5-10. Card Reader Status Word Format

Table 5-15. Card Reader Status Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Unit On Line	Indicates the card reader is connected to the channel and power is applied to the unit.
1	Input Buffer Full	Indicates the input data buffer contains a word ready for transfer to the computer. This condition triggers the input interrupt when that interrupt is enabled.
2	Reader Trouble	Indicates a problem exists within the card reader. This indication may result from a pick failure, card motion error, a light current or dark current error, the hopper being empty, or the stacker being full. This condition triggers the trouble interrupt when that interrupt is enabled.
3	Hopper Empty	Indicates the card reader supply hopper is empty. This condition also sets the reader trouble interrupt (bit 2).
4	Stacker Full	Indicates the card reader stacker is full. This condition also sets the reader trouble interrupt (bit 2).
5	Non Pick	Indicates a failure in the pick mechanism. This could result from an out-of-tolerance card, damaged leading edge, or improper loading of the hopper. This condition sets the reader trouble interrupt (bit 2).
6	Command Buffer Full	Indicates command data has been received by the card reader controller and the unit is busy.
7-23	Spare	Reserved for future expansion.

5-7.5 Data Transfer

The Models 6005-1 and 6005-2 Card Readers generate a 12-bit character by sensing the presence or absence of a hole(s) in each of the twelve rows of each column of a punched card. Each row in a card column corresponds to a specific bit in the computer word. Figure 5-11 illustrates the relationship between card row and computer bit. Also shown in the examples are several data characters, their octal representation, and the corresponding binary word. Each "1" in the binary word represents a punched hole in the card. Comparing the "1s" in a binary word to the corresponding card row will determine which row(s) in a column will contain a punched hole.

Computer Word Bit	11	10	9	8	7	6	5	4	3	2	1	0
Corresponding Card Row	12	11	0	1	2	3	4	5	6	7	8	9
Character A (4400_8)	1	0	0	1	0	0	0	0	0	0	0	0
Character Z (1001_8)	0	0	1	0	0	0	0	0	0	0	0	1
Character + (4012_8)	1	0	0	0	0	0	0	0	1	0	1	0
Character @ (0042_8)	0	0	0	0	0	0	1	0	0	0	1	0

Figure 5-11. Bit Sequence For Data Characters

Each time a card column is read, a character is generated and loaded into the input data buffer. The character is subsequently transferred to the computer as the least significant half (bits 0-11) of the 24-bit data word.

5-7.6 Characteristics

The major operating characteristics of the Models 6005-1 and 6005-2 Card Readers are listed below. Differences between the two models are identified.

Reading Speed (cards per minute)	400 cards nominal (Model 6005-1) 1000 cards nominal (Model 6005-2)
Card Characteristics	Standard 80-column Cards (Per EIA Std. RS292)
Supply Hopper Capacity	500 cards (Model 6005-1) 1000 cards (Model 6005-2) can be replenished during operation
Stacker Capacity	500 cards (Model 6005-1) 1000 cards (Model 6005-2)
Read Sensing	Photoelectric

5-7.7 Character Set

A list of the characters capable of being read by the card reader is provided in Table 5-16. A typical punched card containing each character in the set is provided on Figure 5-12.

Table 5-16. Card Reader Character Set

Symbol	Punched Rows	Octal Equiv.	Symbol	Punched Rows	Octal Equiv.
A	12-1	4400	@	8-4	0042
B	12-2	4200	|	12-8-7	4006
C	12-3	4100	¬	11-8-7	2006
D	12-4	4040	—	0-8-5	1022
E	12-5	4020	SPACE	BLANK	0000
F	12-6	4010	!	11-8-2	2202
G	12-7	4004	"	8-7	0006
H	12-8	4002	#	8-3	0102
I	12-9	4001	$	11-8-3	2102
J	11-1	2400	%	0-8-4	1042
K	11-2	2200	&	12	4000
L	11-3	2100	'	8-5	0022
M	11-4	2040	(	12-8-5	4022
N	11-5	2020	)	11-8-5	2022
O	11-6	2010	*	11-8-4	2042
P	11-7	2004	+	12-8-6	4012
Q	11-8	2002	,	0-8-3	1102
R	11-9	2001	-	11	2000
S	0-2	1200	.	12-8-3	4102
T	0-3	1100	/	0-1	1400
U	0-4	1040	:	8-2	0202
V	0-5	1020	;	11-8-6	2012
W	0-6	1010	<	12-8-4	4042
X	0-7	1004	=	8-6	0012
Y	0-8	1002	>	0-8-6	1012
Z	0-9	1001	?	0-8-7	1006
0	0	1000			
1	1	0400			
2	2	0200			
3	3	0100			
4	4	0040			
5	5	0020			
6	6	0010			
7	7	0004			
8	8	0002			
9	9	0001			

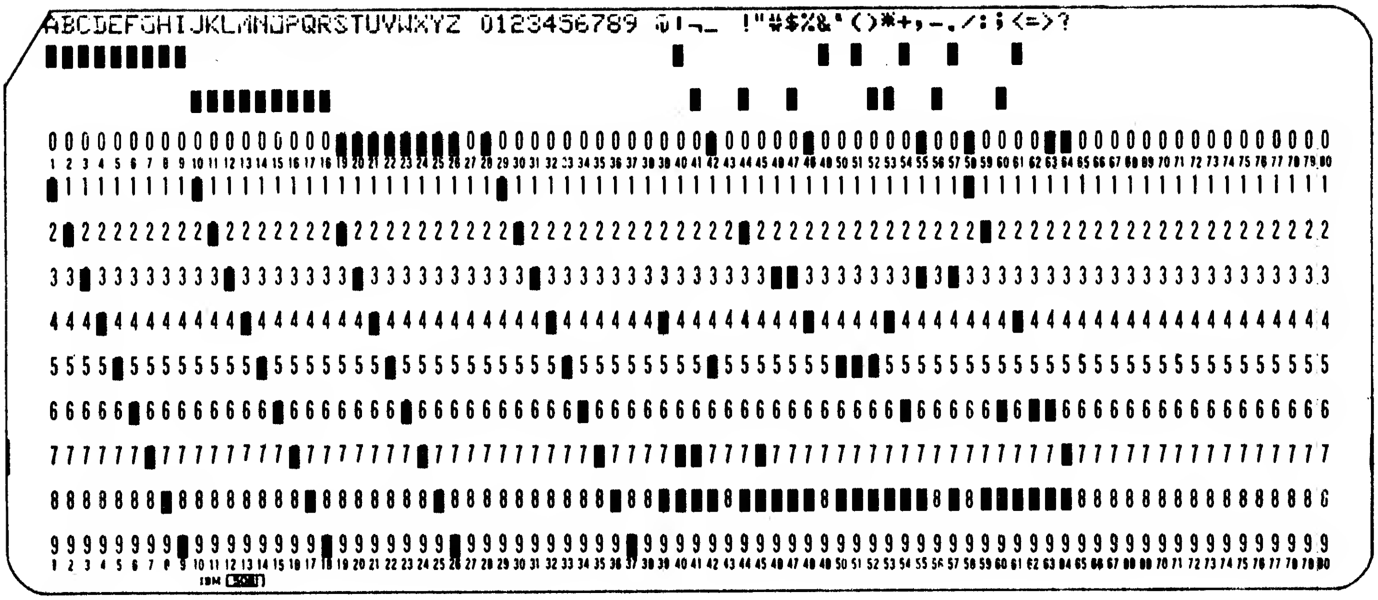


Figure 5-12. Typical Punched Card

5-8 DISC STORAGE SYSTEM MODELS 6006-1, 6006-2 AND 6006-3

5-8.1 General

The disc storage system reads and writes data on a disc pack for permanent or temporary storage. The disc storage system consists of a Disc Controller Model 6006-1 and an associated disc drive unit. Two different drive units may be employed according to storage requirements. The Model 6006-2 Disc Drive Unit is a Memorex Model 660 with a 28-million byte storage capacity. Model 6006-3 contains 7-million bytes of storage capacity and is a Memorex Model 630 Disc Drive Unit.

Data is stored on a disc pack containing 11 aluminum disc coated with a magnetic oxide (Model 6006-2). The inner 20 surfaces store data on 203 concentric circles. Since the 20 surfaces are vertically aligned, each concentric circle group is called a cylinder. Model 6006-3 uses a disc pack with only six aluminum discs with 10 storage surfaces.

5-8.2 Computer Communications

The disc drive system communicates with the computer via the disc controller (installed in the Peripheral Cabinet of the DC 6024 Computer). Each controller communicates with up to eight disc drive units. Communications between the computer and the disc drive system result from program instructions that generate commands for and also monitor the status of the disc drive system.

5-8.3 Command Word

Operation of the disc storage system is initiated by transferring a command word from the computer to the disc storage system. The command word defines the specific operation to be performed and states which drive unit, cylinder, track, and record are to be used for the operation. The command word format is illustrated on Figure 5-13. A brief description of the command word bits is provided in Table 5-17.

CONTROL CODE (SEE TABLE)	DRIVE UNIT NUMBER (0-7)	CYLINDER NUMBER (0-202)	TRACK (HEAD) NUMBER (0-19)	RECORD NUMBER (0-63)
23 22	21 20 19	18 17 16 15 14 13 12 11	10 9 8 7 6	5 4 3 2 1 0

Bit 23	Bit 22	Control Function
0	0	Seek
0	1	Special Action Code
1	0	Initiate ABC Input
1	1	Initiate ABC Output

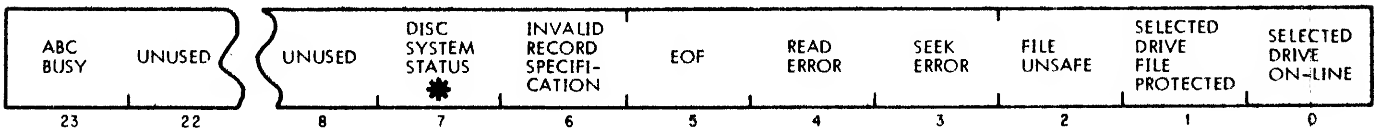
Figure 5-13. Disc Storage System Command Word Format

Table 5-17. Disc Storage System Command Word Bit Description

BIT	FUNCTION	DESCRIPTION
0-5	Record Number (0-63)	Defines that portion of a track containing a block of data.
6-10	Track (Head) Number (0-19)	Specifies a track of the disc pack for reading or writing data. A track is a single plane surface of one disc. There are 20 (0-19) usable tracks in a disc pack.
11-18	Cylinder Number (0-202)	Specifies a cylinder of the disc pack for reading or writing data. A cylinder is a concentric circle on each track aligned vertically. There are 203 (0-202) cylinders per track.
19-21	Drive Unit Number (0-7)	Specifies a drive unit connected to a particular controller.
22-23	Control Code	Determines the function to be performed by the disc storage system. Refer to Figure 5-13 for a list of functions.

5-8.4 Status Word

Operational status of the disc storage system can be determined by the computer at any time. The computer transfers a status request to the system which asks the system for its present status. A status word is formed and returned to the computer. Each bit in the status word is independent and may exist simultaneously. Figure 5-14 illustrates the status word format and Table 5-18 provides a brief description of each status word bit.



*Bit 7 = 0 - Ready
Bit 7 = 1 - Busy

Figure 5-14. Disc Storage System Status Word Format

Table 5-18. Disc Storage System Status Word Bit Description

BIT	FUNCTION	DESCRIPTION
0	Selected Drive On-Line	Indicates the disc drive unit is connected to the channel and power is on.
1	Selected Drive File Protected	Indicates the disc drive unit has been selected for a read only operation.
2	File Unsafe	Indicates the disc drive unit is unsafe for operation.
3	Seek Error	Indicates an error occured during a seek in that an Index slot was detected.
4	Read Error	Indicates an error exists in the read data.
5	End Of File (EOF)	Indicates the disc drive unit has reached the end of the present cylinder (head address 19). This states that no more tracks exist in that cylinder.
6	Invalid Record Specification	Indicates an invalid record has been specified.
7	Disc System Status	Indicates the system is ready for an operation when this bit is set to a logic 0. A logic 1 indicates the system is busy.
8-22	Spare	Reserved for future expansion.
23	ABC Busy	Indicates the Automatic Block Controller is busy.

5-8.5 Data Transfer

The disc storage system communicates with the computer via the controller. The drive unit receives commands and data to be stored from the controller and provides status and read data to the controller. Data is transferred from the computer's input/output structure in blocks controlled by an Automatic Block Controller. Data is transferred in parallel (24 bits) between the computer and the controller and in serial between the controller and the disc storage system.

The disc controller is byte-oriented and receives three 8-bit bytes in a 24-bit data word. Provisions are provided in the controller for word assembly/disassembly.

5-8.6 Characteristics

The major operating characteristics of the disc storage system are listed below.

Capacity

Maximum disc pack capacity, 8-bit bytes	7.25 million (6006-2) 29 million (6006-3)

Data Retrieval Times

Rotational time, milliseconds	25 (2400 RPM± 2%)
Track-to-track positional time, milliseconds	20 maximum
Maximum access time, milliseconds	80
Average access time, milliseconds	50
Data transfer rate, megabits/second	1.25 (nominal, 6006-2) 2.5 (nominal, 6006-3)

Disc Pack Characteristics (6006-2)

Number of recording discs	6
Number of recording surfaces	10
Tracks per surface	200 plus 3 spares
Density, track 000, bits/inch (zero's rate)	765 (nominal)
Density, track 202, bits/inch (zero's rate)	1114 (nominal)

Disc Pack Characteristics (6006-3)

Number of recording discs	11
Number of recording surfaces	20
Tracks per surface	200 plus 3 spares
Density, track 000, bits/inch (zero's rate)	1530 (nominal)
Density, track 202, bits/inch (zero's rate)	2228 (nominal)